



INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification ⁶ : A61B	A2	(11) International Publication Number: WO 99/51138 (43) International Publication Date: 14 October 1999 (14.10.99)
(21) International Application Number: PCT/SE99/00553 (22) International Filing Date: 6 April 1999 (06.04.99) (30) Priority Data: 60/080,430 2 April 1998 (02.04.98) US 09/128,442 3 August 1998 (03.08.98) US (71) Applicant: PRECISE BIOMETRICS AB [SE/SE]; Forsknings- byn Ideon, S-223 70 Lund (SE). (72) Inventors: BERGENEK, Jerker; Lagerbrings väg 4B, S-224 60 Lund (SE). FÄHRAEUS, Christer; Sölvegatan 3, S-223 62 Lund (SE). WIEBE, Linus; Bokhållaregatan 33, S-211 56 Malmö (SE). ÖBRINK, Mårten; Exercisgatan 3, S-211 49 Malmö (SE). (74) Agent: AWAPATENT AB; P.O. Box 5117, S-200 71 Malmö (SE).	(81) Designated States: AE, AL, AM, AT, AT (Utility model), AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CU, CZ, CZ (Utility model), DE, DE (Utility model), DK, DK (Utility model), EE, EE (Utility model), ES, FI, FI (Utility model), GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SK (Utility model), SL, TJ, TM, TR, TT, UA, UG, UZ, VN, YU, ZA, ZW, ARIPO patent (GH, GM, KE, LS, MW, SD, SL, SZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG). Published <i>Without international search report and to be republished upon receipt of that report.</i>	

(54) Title: FINGERPRINT IDENTIFICATION/VERIFICATION SYSTEM

(57) Abstract

A fingerprint identification/verification system using bitmaps of a stored fingerprint to correlate with a bitmap of an input fingerprint, wherein an accurate reference point is located and selected two-dimensional areas in the vicinity of the reference point of the input image of the fingerprint are correlated with stored fingerprint recognition information to determine if the input fingerprint image and the stored fingerprint recognition information are sufficiently similar to identify/verify the input fingerprint.

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania	ES	Spain	LS	Lesotho	SI	Slovenia
AM	Armenia	FI	Finland	LT	Lithuania	SK	Slovakia
AT	Austria	FR	France	LU	Luxembourg	SN	Senegal
AU	Australia	GA	Gabon	LV	Latvia	SZ	Swaziland
AZ	Azerbaijan	GB	United Kingdom	MC	Monaco	TD	Chad
BA	Bosnia and Herzegovina	GE	Georgia	MD	Republic of Moldova	TG	Togo
BB	Barbados	GH	Ghana	MG	Madagascar	TJ	Tajikistan
BE	Belgium	GN	Guinea	MK	The former Yugoslav Republic of Macedonia	TM	Turkmenistan
BF	Burkina Faso	GR	Greece	ML	Mali	TR	Turkey
BG	Bulgaria	HU	Hungary	MN	Mongolia	TT	Trinidad and Tobago
BJ	Benin	IE	Ireland	MR	Mauritania	UA	Ukraine
BR	Brazil	IL	Israel	MW	Malawi	UG	Uganda
BY	Belarus	IS	Iceland	MX	Mexico	US	United States of America
CA	Canada	IT	Italy	NE	Niger	UZ	Uzbekistan
CF	Central African Republic	JP	Japan	NL	Netherlands	VN	Viet Nam
CG	Congo	KE	Kenya	NO	Norway	YU	Yugoslavia
CH	Switzerland	KG	Kyrgyzstan	NZ	New Zealand	ZW	Zimbabwe
CI	Côte d'Ivoire	KP	Democratic People's Republic of Korea	PL	Poland		
CM	Cameroon	KR	Republic of Korea	PT	Portugal		
CN	China	KZ	Kazakstan	RO	Romania		
CU	Cuba	LC	Saint Lucia	RU	Russian Federation		
CZ	Czech Republic	LI	Liechtenstein	SD	Sudan		
DE	Germany	LK	Sri Lanka	SE	Sweden		
DK	Denmark	LR	Liberia	SG	Singapore		
EE	Estonia						

FINGERPRINT IDENTIFICATION/VERIFICATION SYSTEM

This application claims the benefit of U.S. Provisional Application No. 60/080,430 filed April 2, 1998. The Provisional Application is incorporated by reference and is attached hereto as Appendix A.

BACKGROUND OF THE INVENTION

This invention relates generally to the field of fingerprint identification/verification systems. More particularly, this invention relates to a fingerprint identification/verification system using two dimensional bitmaps instead of traditional feature extraction.

Two types of matching applications are used for fingerprints. One-to-one verification is used to compare a fingerprint with either a particular template stored on, for example, a smart card, or a template recovered from a database by having the person provide his or her name, Personal Identification Number (PIN) code, or the like. One-to-many identification is used to compare a fingerprint to a database of templates, and is required when a person presents only his or her finger which is then compared to a number of stored images.

Traditional fingerprint identification by feature extraction has been used by institutions like the Federal Bureau of Investigation (FBI) for identifying criminals and is the most common fingerprint identification system. In feature extraction, the pattern of a fingerprint is checked for any special 'features' such as ridge bifurcations (splits) and ridge endings amongst the meandering ridges of the fingerprint. Once each such feature is identified, the location, that is, the distance and direction between the features, and perhaps

the orientation of each feature, is determined. By storing only the feature location information, a smaller amount of data can be stored compared to storing the complete fingerprint pattern. However, by extracting and storing only the location of each feature, that is, the one-dimensional point on the fingerprint where the feature is located and, perhaps, the type of feature, information for security purposes is lost because all of the non-feature information is then unavailable for comparisons (matching).

Also, in order to determine the absolute location of the features, an unambiguous starting point (reference point) is selected for the fingerprint. Traditional methods locate a 'core point' as the reference point. This core point is usually selected according to different criteria depending on the type of fingerprint, for example, whorl, circular or other type. Thus, a fingerprint in such a traditional system must first be classified as a known type before the core point can be determined and the features located.

Another difficulty encountered with automated fingerprint identification or verification systems is the inability of the system to differentiate between a real fingerprint, that is, a fingerprint on a finger, and an image or plastic model of a fingerprint. In traditional systems the type of sensor can help, for example, a heat sensor to detect body heat, but these sensors can be defeated.

In addition, identification presents difficulties when the database of possible fingerprints becomes quite large. In traditional fingerprint systems, each type of fingerprint is categorized and the types of features provide additional subclasses. Nevertheless, the number of classes and subclasses is quite small when compared to the number of fingerprints which may be in any particular class or subclass. Also, once a class or subclass is selected, possible matches in a different

class or subclass of the same level of the hierarchy are not checked. Thus, for fingerprints which do not clearly fall within a particular class or subclass, there may be stored fingerprints in the database which are not checked.

- 5 Accordingly, a search for a matching fingerprint image on file can be both time consuming and result in a false indication that the particular fingerprint is not on file.

OBJECTS AND SUMMARY OF THE INVENTION

An object of the present invention is to provide a fingerprint identification system which identifies fingerprints more accurately than prior systems.

5 Another object of the present invention is to identify fingerprints by comparing entire two dimensional regions of fingerprint images rather than just the locations of features.

10 An additional object of the present invention is to accurately and efficiently find a reference point in the image from where to start the identification or verification process.

A further object of the present invention is to determine dynamics of the fingerprint as the image is being made to differentiate a true fingerprint from a false/fake fingerprint placed on the sensor.

15 Another object of the present invention is to establish a non-hierarchical database which allows for rapid matching of a candidate fingerprint and matching without requiring that the candidate fingerprint belong to a particular class.

20 A further object of the invention is to provide a fingerprint processing method, and a device for accomplishing the method, having the steps of: (1) obtaining an image of a fingerprint comprising ridges and valleys; (2) searching the image to locate a reference point; and (3) selecting the reference point and a region in the vicinity of the reference point as a recognition template for the image. This method can have the following additional steps: (1) applying the fingerprint to a scanning device; (2) scanning the fingerprint to generate an image signal; and (3) storing the image signal as a digital image. In addition, this method can include any or all of the following sub-methods:

30 (A) - (1) vectorizing the digital image; (2) selecting a starting sub-area in the vectorized image; (3) scanning from the starting sub-area along an orientation of each subsequent

sub-area to locate a first sub-area having a horizontal orientation, the first sub-area included in a first horizontal structure; (4) scanning from the first sub-area across acceptable structures and along a path of acceptable sub-areas until an unacceptable sub-area is located; and (5) selecting the center point of the last scanned acceptable sub-area as the reference point;

(B) - (1) calculating the geographic center of the digital image; and (2) selecting the geographic center as the reference point; or

(C) - (1) binarizing the digital image; (2) determining the row of the digital image which has the greatest number of binary transitions; (3) determining the column of the digital image which has the greatest number of binary transitions; and (4) selecting a point in the image by following a path starting from a point in the image having the row and the column as coordinates.

Also, the searching step of this method can include the following steps: (1) selecting a starting point; (2) following along at least one ridge proximate the starting to locate a ridge of a first type; (3) selecting adjacent ridges of the first type along a predetermined path to locate a ridge of a second type; and (4) selecting a point on the last located ridge of the first type as the reference point.

In addition, the selecting step of this method can include the following steps: (1) selecting the region to include the reference point, the region having a size and a shape; (2) storing the recognition template; (3) selecting the region to include the reference point; (4) selecting other regions, each of the other regions having a respective size and a respective shape, each such other region located with respect to the reference point according to relative location information; and

(5) selecting the other regions and the respective relative location information for each respective other region as part of the recognition template for the image.

In addition, this method can include the following steps:

5 (1) storing the recognition template; (2) encrypting one or more of the region, the other regions, and the relative location information; and (3) compressing one or more of the region, the other regions, and the relative location information.

10 An additional object of this invention is to provide a fingerprint matching method, and a device for accomplishing the method, having the steps of: (1) obtaining an image of a fingerprint comprising ridges and valleys; (2) searching the image to locate a reference point; (3) selecting the reference
15 point and a region in the vicinity of the reference point; (4) selecting at least one recognition template, each recognition template comprising a template reference point and a template region; (5) correlating at least a portion of the region with the template region to generate a correlation result; and (6)
20 determining whether the correlation result exceeds a predetermined matching requirement. This method can also include the following steps: (1) obtaining from the recognition template, relative location information of at least one other template region; (2) selecting another region from the image
25 utilizing the relative location information with respect to the template reference point; (3) correlating at least a portion of the another region with the other template region to generate a correlation result; and (4) determining whether the correlation result exceeds a predetermined matching requirement.

30 Another object of this invention is to provide a fingerprint processing method, and a device for accomplishing the method, having the following steps: (1) obtaining sequential multiple images of a fingerprint comprising ridges

and valleys; (2) determining dynamics of the obtaining step procedure by comparing the multiple images to each other. The method can also have the following additional step: determining from the dynamics if the fingerprint is real.

5 A further object of this invention is to provide a fingerprint information storage method, and a device for accomplishing the method, having the following steps: (1) obtaining from a fingerprint values for each of a number of fingerprint characteristics; (2) assigning each of the values to a respective coordinate, the coordinates defining a point in a dimensional space having the number of dimensions; and (3) associating information concerning the fingerprint with the point. This method can also include the step of locating information concerning other fingerprints based on the
15 proximity of other points in the dimensional space associated with other respective fingerprints.

Also, an object of the present invention is to provide a fingerprint processing device, and a method for operating the device, having: (1) a sensor for detecting a fingerprint and
20 for generating an image signal corresponding to the fingerprint; (2) a processor for receiving the image signal and for identifying a reference point and a region in the vicinity of the reference point in an image formed from the image signal; and (3) a storage device for storing information
25 concerning the reference point and a portion of the image. This device may also include a correlator for comparing information received from the storage device and information concerning the region in the vicinity of the reference point.

30 An additional object of the present invention is to provide a storage template, and a method for creating the template, for a fingerprint processing system having: (1) a first region bitmap; (2) a reference point location; (3)

outlying region bitmaps; and (4) relative location information, the relative location information corresponding to the location of each of the outlying region bitmaps with respect to the reference point location.

- 5 These objects and other objects, advantages, and features of the present invention will become apparent to those skilled in the art upon consideration of the following description of the present invention.

BRIEF DESCRIPTION OF THE DRAWINGS

Fig. 1 is a flow diagram illustrating an enrollment process according to an embodiment of the present invention;

Fig. 2 is a binarized version of a captured image according to one embodiment of the present invention;

Fig. 3 is a vectorized version of the same captured image which is binarized in Fig. 2 according to one embodiment of the present invention;

Fig. 4 illustrates the possible sub-area orientations according to an embodiment of the present invention having eight possible orientations;

Fig. 5 illustrates the acceptable roof structures according to one embodiment of the present invention;

Fig. 6 illustrates the candidate sub-areas during a downward search according to one embodiment of the present invention;

Fig. 7 illustrates the possible acceptable left endpoints for an acceptable horizontal line structure according to one embodiment of the present invention;

Fig. 8 illustrates the possible acceptable right endpoints for an acceptable horizontal line structure according to one embodiment of the present invention;

Fig. 9 is a flow diagram illustrating a first horizontal line structure search according to one embodiment of the present invention;

Fig. 10 is a flow diagram illustrating a downward search for the reference point according to one embodiment of the present invention;

Fig. 11 is a flow diagram illustrating the scan of a structure to determine if the structure is acceptable according to one embodiment of the present invention;

Fig. 12 illustrates the center region, outlying regions, and the location vectors for a recognition template according to one embodiment of the present invention;

Fig. 13 is a flow diagram illustrating the matching process according to one embodiment of the present invention;

Fig. 14 illustrates the matching procedure for both the center regions and the outlying regions according to one embodiment of the present invention;

Fig. 15 illustrates the increasing coverage as a fingerprint is dynamically sensed to form an image according to one embodiment of the present invention;

Fig. 16 illustrates an ulnar loop structure of a fingerprint;

Fig. 17 illustrates a radial loop structure of a fingerprint;

Fig. 18 illustrates an whorl structure of a fingerprint;

Fig. 19 illustrates an arch structure of a fingerprint;

Fig. 20 illustrates a tented arch structure of a fingerprint;

Fig. 21 illustrates a defect/scared structure of a fingerprint;

Fig. 22 illustrates micro-features (minutiae) of a fingerprint;

Fig. 23 is a table illustrating the different kinds of minutia found in a fingerprint;

Fig. 24 illustrates a TouchViewII fingerprint scanner;

Fig. 25 illustrates a Verdicom fingerprint sensor;

Fig. 26 is a schematic diagram of a fingerprint scanner;

Fig. 27 illustrates an original fingerprint image;

Fig. 28 is a histogram showing a mean thresholding value of 200;

Fig. 29 illustrates the original fingerprint image of Fig. 27 after being subjected to global thresholding;

Fig. 30 illustrates a matrix of marked pixels according to a thresholding method using a ridge-valley orientation detector;

Fig. 31 illustrates a binarised image;

5 Fig. 32 illustrates a matrix of marked pixels according to a thresholding method using an extended ridge-valley orientation detector;

Fig. 33 illustrates an original fingerprint image;

Fig. 34 illustrates the original fingerprint image of Fig. 33 after being subjected to global thresholding at a mean thresholding value of 150;

Fig. 35 illustrates the original fingerprint image of Fig. 33 after being subjected to thresholding using the extended ridge-valley orientation detector;

15 Fig. 36 illustrates the result when a majority operation is performed on the fingerprint image in Fig. 35 six times;

Fig. 37 illustrates the skeleton of the fingerprint image shown in Fig. 36;

Fig. 38 illustrates a ridge-ending in a fingerprint image;

20 Fig. 39 illustrates a ridge-ending in a fingerprint image;

Fig. 40 illustrates a core point and a delta point for a fingerprint image;

Fig. 41 illustrates a qualitative Gaussian surface;

25 Fig. 42 illustrates a directional filter for an arbitrary angle ϕ ;

Fig. 43 illustrates a vectorised image created using a derived Gaussian filter;

Fig. 44 illustrates a vectorised image created using a derived Gaussian filter;

30 Fig. 45 illustrates a matrix of pixel values across a ridge/valley;

Fig. 46 illustrates an image of the matrix shown in Fig. 45;

Fig. 47 illustrates a matrix of pixel values having the optimal size for a 500 dpi image when using the minimum absolute difference;

Fig. 48 illustrates an original fingerprint image;

5 Fig. 49 illustrates vectorisation of the image in Fig. 48 by use of a Gaussian derivative filter;

Fig. 50 illustrates vectorisation of the image in Fig. 48 by using minimum absolute difference;

Fig. 51 illustrates a core point on a fingerprint image;

10 Fig. 52 illustrates a frequency analysis of a fingerprint, showing dominant frequencies for all rows and columns using the FFT routine;

Fig. 53 illustrates a frequency analysis of a fingerprint, showing dominant frequencies for all rows and columns using the
15 FFT routine;

Fig. 54 illustrates nine samples of the same fingerprint, with arrows indicating reference points defined by the FFT routine;

20 Fig. 55 illustrates a visualization of the ridge count method;

Fig. 56 illustrates nine samples of the same fingerprint, with arrows indicating reference points defined using the ridge count routine;

Fig. 57 illustrates an enlarged fingerprint image B;

25 Fig. 58 illustrates a fingerprint image A;

Fig. 59 is a surface plot comparing bitmaps A and B;

Fig. 60 illustrates the best-matched position between images B and A of respective Figs. 57 and 58;

Fig. 61 illustrates an enlarged fingerprint image C;

30 Fig. 62 is a surface plot illustrating the matching of images A and C of respective Figs. 58 and 61; and

Fig. 63 illustrates a three-dimensional plot of the match scores for eight fingerprints matched against each other.

DETAILED DESCRIPTION OF THE INVENTION

While this invention is susceptible of embodiment in many different forms, the drawings show and the specification herein describes specific embodiments in detail. However, the present disclosure is to be considered as an example of the principles of the invention and is not intended to limit the invention to the specific embodiments shown and described. In the description below, like reference numerals are used to describe the same, similar or corresponding parts in the several views of the drawing.

The present invention is described below in two main sections: (1) an enrollment procedure and device; and (2) a matching procedure and device. The matching procedure and device section also describes the use of finger dynamics and a non-hierarchical database as employed in the present invention.

Fig. 1 illustrates a procedure for selecting information to be stored as a template by enrollment 100, for example to register authorized people, according to one embodiment of the present invention. In this embodiment, the captured image is a digital image. A binarized version of the captured image is illustrated in Fig. 2. This binarized image is organized into an orthogonal grid 200 having rows 210 and columns 220 of picture elements or pixels. Each pixel is encoded to a gray-scale value. The rows 210, the horizontal orientation, are numbered in increasing order moving down from the part of the image corresponding to the part of the fingerprint closest to the fingertip; and the columns 220, the vertical orientation, are numbered in increasing order from left to right. Also, the terms 'up', 'down', 'left', 'right', and variations thereof, are used in this specification to refer to the top (lower row numbers), bottom (higher row numbers), leftside (lower column numbers), and rightside (higher column numbers), in an image,

respectively. However, the present invention can be implemented using other types of images and image organizations, such as for example, a hexagonal grid or an analog image.

5 The enrollment procedure 100 according to one embodiment of the present invention is described below with respect to each step of the procedure.

Image Capture 110: The first step in enrollment 100 is to capture the image with an image capturing device or sensor.
10 The sensor can be, for example, a heat sensor, a light sensor, an optical sensor, a silicon sensor, or any other technology used to capture a fingerprint image. The sensor is coupled to a signal processor and/or microprocessor with sufficient read only memory (ROM) and random access memory (RAM) for operating
15 on the image signal produced by the sensor.

If high security is required, such as for access to high-security computer network, the enrollment process 100 could be monitored while the person's fingerprint is placed on the sensor to ensure a high quality image is captured for storage
20 as a template. Lower security, such as for access to an automatic teller machine (ATM) lobby, however, does not require as much, if any, supervision during enrollment 100 since a lower quality template can be tolerated.

Quality Check 120: The fingerprint image is checked for
25 dryness or wetness. If the image is 'too dry' the pressure applied to the sensor was too light or the sensor failed to detect parts of ridges because of fingertip dryness. If the image is 'too wet', moisture on the finger 'flooded' the fingerprint valleys. Wetness or dryness is detected by
30 analyzing the image for too few dark pixels (dryness) or, too many dark pixels and continuous dark areas (wetness). If the image is rejected, the person is asked to correct the problem and another image is taken.

5 Binarization 130: Once an image of the appropriate quality is captured 110, 120 the gray-level image is converted into a black-and-white (binarized) image, see Fig. 2, of the sensed fingerprint. This binarization is extremely sensitive to the quality of the image. Binarization 130 is performed using a gray-scale threshold. Thus, for example, a pixel having a gray-scale value above a threshold value is determined to be black, and a pixel having a gray-scale value level below the threshold value is determined to be white. The threshold value can be global (the same threshold value is used for the entire image), or local (different threshold values are calculated separately for different areas of the image).

15 Also, in one embodiment of the present invention, to aid in binarization 130, information from the ridge/valley directions are used to enhance the binarized image. For example, an isolated pixel which has a gray-scale value just high enough to be considered black and thus, part of a ridge, will instead be set to white if all the surrounding pixels are considered to be in a valley. This enhancement is particularly
20 useful for lower quality or noise-affected images. Another embodiment of the present invention combines both local thresholds and ridge/valley direction information from the same area as part of binarization 130.

25 Restoration 140: Restoration is similar to, and is interconnected with, binarization 130. However, restoration 140 is performed after binarization 130. Basically, restoration 140 takes advantage of knowledge of how fingerprints are known to appear, for example, the generally continuous nature of fingerprint ridges. Techniques such as
30 the use of local ridge/valley directions described above are also used for restoration 140. Another restoration technique determines a pixel's value based on the particular combination of the neighboring pixel values. Other restoration methods

consider and restore the image based on expected ridge/valley widths and other physical fingerprint characteristics.

Reference Point Determination 150: After the image is binarized 130 and restored 140, a reference point for the image must be selected. Finding a repeatedly locatable reference point has traditionally been extremely complex because of the many different types of fingerprints. Even in traditional manual fingerprint classification, which has a large set of rules for identifying the reference points for many types of fingerprints, reference points cannot be defined for some types of fingerprints.

However, in one embodiment of the present invention only two procedures are required. The first procedure 152 is based on a vectorization of the gray-scale image. The second procedure 154, which is used only if the first procedure 152 is unable to locate a reference point, locates the geographic center of the image. Alternatively, the second procedure 154 can be based on counting the ridges in a binarized image, or by calculating fast Fourier transforms (FFTs) of the fingerprint image and selecting the point corresponding to the dominant frequencies.

The first procedure 152 locates a reference point from a vector representation of the gray-scale image, that is, a vectorized image 300. Fig. 3 illustrates such a vectorized image. Vectorization 152 is performed by dividing the image into sub-areas and by assigning an orientation to each sub-area 305. Fig. 4 illustrates the possible sub-area orientations according to the embodiment of the present invention shown in Fig. 3. With this first procedure 152, the reference point is defined as either the center pixel of the last of the leftmost of two sub-areas of the image defining a 'roof' structure, or the center pixel of the last middle (or, if there are two middle sub-areas, the left middle) sub-area 360 of a horizontal

line structure which is encountered when searching downward from the top of the vectorized image 300. Fig. 5 illustrates the acceptable roof structures. Basically, a roof structure is defined as two sub-areas pointing upwards and askew towards each other, that is, 2, 3 or 4 as a left sub-area and 6, 7 or 8 as a right sub-area. Fig. 6 illustrates an acceptable horizontal line structure according to one embodiment of the present invention. Also, Figs. 7 and 8 illustrate acceptable left and right endpoints, respectively, for an acceptable horizontal line structure according to one embodiment of the present invention. The acceptable left endpoint patterns shown in Fig. 7 have orientation numbers are 2; 3; 1 followed to the left by a 2, 3 or 4; 4 followed to the right by a 4; or 4 followed to the left by a 1. The acceptable right endpoint patterns shown in Fig. 8 have orientation numbers are 7; 8; 1 followed to the right by a 6, 7 or 8; 6 followed to the left by a 6; or 6 followed to the right by a 1.

Most fingerprints have roof structure ridges below multiple horizontal ridges which gradually increase in curvature towards the center of the fingerprint until a ridge is so curved as not to be considered either a roof structure or a horizontal line structure. In other words, the reference point located with this first procedure 152 is the topmost point of the innermost upward curving ridge, that is, where the ridge almost curves, or does curve, back on itself.

To locate the reference point in the vectorized image 300, the first procedure 152 begins by searching for a first horizontal line structure with endpoints having orientations pointing upwards and inwards. Then, the procedure 152 searches downward until acceptable horizontal line structures and roof structures give way to other types of, though usually almost vertical, structures. Should this transition from horizontal line structures and roof structures not be found, the reference

point sub-area 360 is presumed to have been missed. The first procedure 152 indicates that the downward search has passed the reference point when the acceptable horizontal line structures begin to lengthen again, that is, become much longer. While searching upwards, the scan searches for a roof structure as in the downward search, but continues the search until the next horizontal line structure is encountered before selecting the reference point.

The reference point located according to the first procedure 152 is stable over any number of images of the same fingerprint while also being located in an area with a high degree of information content, that is, an area with little redundant information such as parallel ridges. This location in a high information area aids in the matching procedure. Furthermore, this procedure locates the same reference point even if the fingerprint is presented at different angles with respect to the sensor. For example, the same reference point will be located even if one image of the fingerprint is rotated +/- 20 degrees with respect to another image of the same fingerprint.

Locating the reference point is repeated for a multiple number of images of the same fingerprint to verify that the reference point is stable over these images and to ensure that when the fingerprint is later imaged for identification/verification, the same reference point is located. In one embodiment, ten images were found sufficient.

While the present invention can operate with a vectorization using N orientations, with a minimum of N=2, the embodiment illustrated in Fig. 3, has eight possible orientations that is, N=8. In the embodiment shown in Fig. 3, each vector represents the predominant orientation of an 8 pixel by 8 pixel sub-area of the image. The size of the sub-area used for selecting an orientation generally corresponds to

the resolution of the image. For example, an 8 pixel by 8 pixel sub-area is sufficient for a digital image of 500 dots per inch resolution. In Fig. 3, the eight orientations are evenly spaced but the direction of the orientations is not distinguished. For example, the vectors of 90 degrees and 270 degrees have the same orientation.

As illustrated in Fig. 4, each of the orientations can be assigned a number:

<u>Vector (degrees)</u>	<u>Orientation Number</u>
90 and 270 (vertical)	1
67.5 and 247.5	2
45 and 225 (left oblique)	3
22.5 and 202.5	4
0 and 180 (horizontal)	5
157.5 and 337.5	6
135 and 315 (right oblique)	7
112.5 and 292.5	8
non-defined, background	0

Most conventional vectorization methods produce a good representation of the original image once the thresholds for the foreground and background of the image are determined. To define this boundary, in one embodiment of this invention and as illustrated in Fig. 3, boundaries of the vector image foreground are set according to the following rules, applied in order:

1. The orientation at the bottom of every column is vertical 370;

2. The orientation at the top of every column is horizontal 375;

3. The rightmost orientation of every row is right oblique 380; and

4. The leftmost orientation of every row is left oblique 385.

These boundary conditions allow the search for a reference point to start virtually anywhere in the vectorized image and iteratively follow a set procedure to locate the same reference point.

The downward search according to one embodiment of the present invention is described in further detail below, as Steps A, B, C and D and with reference to Figs. 3-11.

Step A. (Start): Start at any sub-area in the foreground of the vectorized image. In one embodiment, the starting point 310 is the intersection of the vertical column of the geographic center of the image, and the horizontal row of one-third of the way to the top of the image from the geographic center.

Step B. (Search for first horizontal line structure): Search by following the orientation of each sub-area in the image generally upwards from sub-area to sub-area until a first horizontal line structure 320 is encountered. A first horizontal line structure 320 has a left endpoint 330 with an orientation number of 2, 3 or 4 and a right endpoint 340 with an orientation number of 6, 7 or 8. This first horizontal line structure search 500 is illustrated in Fig. 9 and is performed as follows:

<u>Current Sub-area</u>	<u>Next Sub-area</u>
1, 2 or 8	move up one row
3 or 4	move up one row, move right one column
5	perform a left endpoint search for a first horizontal line structure
6 or 7	move up one row, move left one column
0	move down ten rows

Orientation number 0 means the current sub-area is in the background 350 of the image which means that the search has moved too far up in the image. Therefore, the search moves ten rows downward before continuing. When a sub-area with a horizontal orientation, that is orientation number 5, is encountered, a search is made to determine if the first horizontal line structure has been found. If no first horizontal line structure is found after, for example, 100 iterations of Step B, this first procedure 152 has failed to locate a reference point, and the second procedure 154 is used.

The left endpoint search 510 for a first horizontal line structure is performed as follows:

<u>Current Sub-area</u>	<u>Next Sub-area</u>
1, 6, 7, 8 or 0	move left one column, return to first horizontal line structure search
2, 3 or 4	move right one column, perform right endpoint search for first horizontal line structure
5	move left one column

The right endpoint search 520 for a first horizontal line structure is performed as follows:

<u>Current Sub-area</u>	<u>Next Sub-area</u>
1, 2, 3, 4 or 0	move right one column, return to first horizontal line structure search
5	move right one column
6, 7, 8	begin downward search

Step C. (Downward Search): Searches downwards from the first horizontal line structure 320 until the reference point is found, or the search has skipped the reference point. A skipped reference point is indicated by the length of the

acceptable horizontal line structures because above the reference point the acceptable horizontal line structures get smaller in the downward direction, but below the reference point the acceptable horizontal line structures get longer in the downward direction. This downward search procedure is illustrated in Fig. 10. Roof structures, as illustrated in Fig. 5, can be considered the shortest acceptable horizontal line structures and are acceptable structures. Also, while the first horizontal line structure 320 is a type of acceptable horizontal line structure, acceptable horizontal line structures encompass a greater degree of variation, see Figs. 6 and 11.

The first step in the downward search is to determine the length 810 of the current acceptable structure 600 by counting the number of sub-areas of the acceptable structure. Then, as illustrated in Figs. 6, 10 and 11, select 820 the middle sub-area 605 of the acceptable structure as the possible reference sub-area and investigate 830 the following candidate sub-areas, in the following order: (1) down one row 610; (2) down one row, left one column 620; (3) down one row, right one column 630; (4) down one row, left two columns 640; (5) down one row, right two columns 650.

If any of these candidate sub-areas are part of an acceptable structure 845,847 select this acceptable structure 850 for determining the next middle sub-area for the next iteration of step C. However, if the length of the acceptable structure 600 is much longer, for example six times longer, than the shortest length of the acceptable structures encountered so far 815, the reference point is considered to have been skipped and an upward search needs to be performed 860, see Step D.

If no acceptable structure, that is, a horizontal line or a roof structure, has been located among the candidate sub-

areas 847, the possible reference sub-area is, in fact, the actual reference sub-area 360, and the center pixel of the actual reference sub-area is the reference point.

5 The acceptable horizontal line structure search 846 is performed as follows:

<u>Current Sub-area</u>	<u>Next Sub-area</u>
1, 2, 3, 7, or 8	select next candidate sub-area
4, 5 or 6	perform acceptable left endpoint search

The acceptable left endpoint search 882, 884 is performed as follows:

<u>Current Sub-area</u>	<u>Next Sub-area</u>
4, 5 or 6	move left one column,
	check for acceptable left endpoint
1, 2, 3, 7, or 8	select next candidate sub-area

If an acceptable left endpoint is found, the acceptable right endpoint search 886, 888 is performed as follows:

<u>Current Sub-area</u>	<u>Next Sub-area</u>
4, 5 or 6	move right one column, check for acceptable right endpoint
1, 2, 3, 7, or 8	select next candidate sub-area

25 If both an acceptable right endpoint and an acceptable left endpoint are found 892, the horizontal line structure is acceptable and the middle sub-area of this acceptable horizontal line structure is used to determine the next candidate sub-areas.

30 Step D. (Upward Search) Searches upwards according to similar rules as Step C, except the search for acceptable structures is performed in the upward directions.

35 Thus, according to one embodiment of the present invention, a stable reference point can be identified by locating the first point in the fingerprint image, scanning

downward, which has a greater curvature than even the roof structures, for example, a left sub-area orientation of 1 and a right sub-area orientation of 8. Since the structures above this point are common to virtually all kinds of fingerprints, that is, primarily parallel meandering ridges, finding a starting point and then searching downwards will almost always locate a stable reference point.

The second procedure 154, according to one embodiment of the present invention, is used to locate the geographic center only when the first procedure 152 fails to locate the reference point.

The geographic center of the binarized fingerprint image 200 is defined as the pixel in the foreground of the image where the same number of pixels are located above the point as below and the same number of pixels are located to the right as to the left. Thus, the foreground of the image must be separately identified from the background.

In one embodiment of the present invention, the boundary of the foreground is determined using the variance of the pixel values. The pixel values only vary slightly over the entire background, whereas in the foreground the pixel values vary significantly because the ridge structures have significant variation between the valleys which, in one embodiment of the present invention, are white and the ridges which, in one embodiment of the present invention, are black. Thus, by calculating the variance of the pixels, the boundary between the foreground and background can be determined.

An alternative procedure for locating the foreground boundary of the image is to find the first pixel of every row and column that corresponds to a part of a ridge when searching toward the center of the binarized image 200 from each edge of the image. In one embodiment of the present invention such a pixel has a value higher than a certain threshold whereas the background has pixels having values below the certain

threshold. Because the ridges are in the foreground, the pixels so located define the boundary of the foreground.

Once the foreground boundary has been determined, the number of foreground pixels in each row and column are counted and the column that has as many foreground pixels to the left as to the right and the row that has as many foreground pixels above as below are selected as the coordinates of the reference point for the image.

An alternative first or second procedure 152, 154 for finding a reference point is based on ridge counting using the binarized, restored image. In this alternative procedure, the number of ridges crossing each vertical and horizontal grid line in the image are determined. The point where the row and the column having the highest respective ridge counts intersect is selected as a starting point. This row is selected as the reference point row. From this starting point, a search follows along three neighboring ridges to the topmost point (lowest row number) and this column is selected as the reference point column. These two steps, are described in greater detail below as Steps A and B.

A. Along each row and column, the search counts all transitions from black to white and white to black. Then the search selects the point (row, column) with the highest ridge count, that is the greatest number of transitions, as a starting point, or if three or more rows/columns having the same ridge count, the middle row/column is selected.

B. Using the row value from the starting point, the search then selects the reference point column by following the ridge closest to the starting point and the two closest neighboring ridges upwards to the respective top points. The average of these three ridge top points is selected as the reference point column.

Recognition Template Selection 160: After the reference point has been determined, both the reference point and a first

region of the binarized image is selected for storage as part of a recognition template. As illustrated in Fig. 12, in one embodiment of this invention, a region centered on the reference point 1120 is selected as a 'center region' 1100. This center region, according to one embodiment of the invention, is a square having a size of 48 pixels by 48 pixels. Also, additional outlying regions 1110 of the binarized image are selected for storage in the recognition template. In one embodiment of the present invention, four to eight outlying square regions 1110 are selected, each outlying region having a size of 48 pixels by 48 pixels. The outlying regions 1110 can be selected to be neighboring, proximate, or in the vicinity of the center region 1100. However, this invention also encompasses center regions and outlying regions of different sizes, shapes and more distant locations. The size, shape and location of the regions can be selected so as to maximize the useful information in accordance with, for example, the number of pixels available from the sensor, or other considerations. The outlying regions 1110 are located with respect to the reference point 1120 using location information 1130 which is also stored in the recognition template. This location information is illustrated in Fig. 12 by vectors 1130 originating at the reference point.

Outlying regions 1110 can be selected based on fixed positions relative to the center region 1100 or reference point 1120, or in one embodiment, the fingerprint binary image can be scanned for features and each of the feature locations can be used as the basis for defining outlying regions. By selecting outlying regions 1110 including features, more information is stored than when outlying regions containing parallel ridges are selected. More information is conveyed in features because features have less redundant information than parallel ridges and, thus, are more easily distinguished when compared. The features are initially located using conventional methods, for

example, following a ridge line to the point where the ridge ends or splits (bifurcates). Once identified, the outlying regions 1110 are selected to include as many feature locations as possible thereby maximizing the amount of useful information being stored. However, if the image lacks a sufficient number of features for the number of outlying regions 1110 required, the remaining outlying regions can be selected using default locations.

Once selected, the reference point 1120, the pixels of the center region 1100, the pixels of the outlying regions 1110, and the location information 1130 are stored in the recognition template. All or part of the recognition template may be compressed and/or encrypted before being stored.

The matching procedure is described below with respect to Figs. 13 and 14. This matching procedure can be used for both identification and verification. If verification is desired, a particular recognition template, such as for example, a template stored on a smart card, is compared to the candidate image information. If identification is required, a search of a recognition template database is performed based on particular characteristics of the candidate image information to locate potential matching recognition templates. Identification, therefore, requires a series of matching procedures.

Image Capture and Dynamics 1202: When a finger is pressed against the sensor to capture one or more images of the candidate fingerprint, the percentage of black pixels change from approximately zero to around 50% of the pixels. In one embodiment, a threshold is used to determine whether a sufficient number of pixels have become black so that matching can be performed.

In one embodiment of the present invention, multiple images of the candidate fingerprint are acquired at the maximum possible speed allowed by the system which is between 5 and 50

images per second while the finger is pressed against the sensor. Each of these candidate images, if time permits, can be processed separately using the matching procedure 1200. By using multiple images, dynamic properties of the fingerprint capture procedure can be sensed. For example, the multiple images can be compared to each other to improve image quality and to verify the stability of the reference point.

One advantage of acquiring images at a rate of at least 5 to 50 times per second (Hertz) is that fake fingerprints can be detected. This detection is possible because the images captured from a real finger are not instantly stable. Fake fingerprints, such as formed from silicone or rubber, or a paper copy of a fingerprint do not exhibit this kind of instability.

For a real fingerprint, as the images 1410 are acquired, the foreground 1420 of each image becomes progressively larger relative to the previous image so that the foreground covers more and more of the sensor area. This increasing coverage follows from the finger being pressed harder and harder against the sensor. Fig. 15 illustrates this process. If a paper copy of a fingerprint is presented the foreground does not gradually increase in size. Instead, with a paper copy, the foreground, which contains the fingerprint structures, instantly occupies a constant percentage of the sensor area. Similarly, the foreground of a fake fingerprint does not necessarily increase in size at the same rate as a real fingerprint. Also, during the first couple of hundreds of milliseconds, the darkness of the image gradually increases as the pressure of the fingertip against the sensor increases. Accordingly, by measuring this change in the median pixel values of the foreground, a distinction can be made between a fake fingerprint and a real fingerprint.

Additionally, when a person places a finger on the scanner, the finger is never completely still, that is, each successive image differs slightly from the next. Thus, a

comparison of the successive images to each other to ensure that none are exactly the same will also indicate that the images being received are from a real fingerprint.

Another use of finger dynamics is to detect the thickness of the ridges as the finger is pressed towards the sensor. As a finger is pressed against a sensor, the ridges appear to thicken because the ridges are 'flattened' against the sensor surface. Additionally, this flattening occurs in a known way. Further, a minute, non-symmetric displacement of the pattern, which can be the result of a small twist or a change of the way of pressing the finger on the sensor is another way of separating real fingerprints from fake fingerprints. Such a difference can be detected by comparing an image with a subsequent one. In both cases, the dynamics of the finger is used as a way to accept real fingerprints and reject counterfeits.

The present invention also encompasses using finger dynamics during image capture for enrollment.

Quality Check 1204: If time permits, a quality check 1204, similar to the quality check 120 for enrollment 100 can be performed on the input candidate image.

Binarization 1208: The candidate image is binarized in the same way as an enrolled image.

Restoration 1210: If time permits, image restoration 1210 similar to the restoration 140 for enrollment 100 can be performed on the input candidate image.

Reference Point Determination 1212: The input reference point 1310 is located in the input candidate binarized image using the same procedure 150 used for the enrolled image.

Input Candidate Center Region Determination 1220: An input center region 1320, in one embodiment, a square have a size of $X+m$ pixels by $X+m$ pixels is selected around the candidate image's reference point 1310. 'X' is the size of the center region stored in the recognition template, and 'm' is selected to be between X divided by 4 ($X/4$) and 2 multiplied by X ($2*X$),

which for the 48 pixel by 48 pixel example is between 12 and 96.

Center Region Correlation 1230: Correlate the center region of the recognition template with a selected portion of the input candidate center region to determine if these center regions match.

Correlation for this invention is meant in its broadest sense, that is, a pixel-by-pixel comparison between portions of the candidate image information and the stored recognition template information. Correlation at its simplest, means that if a pixel in the image matches a pixel in the template, a fixed value, such as "1", is added to a total. If the pixel in the candidate image does not match the pixel in the recognition template, no addition is made to the total. When each pixel in the candidate image and the recognition template have been checked for matching (compared), the total indicates the amount of correlation between the image and the template. Thus, for example in one embodiment, a match value between 0%, that is zero, and 100%, that is one, is obtained from the correlation. 0% indicates a complete mis-match and 100% indicates a perfect match. Of course, other types of correlation are encompassed by this invention, including: (1) multiplying each pixel in the image by the corresponding pixel in the template and integrating to obtain the correlation; and (2) logically 'XOR-ing' (exclusive OR) each pixel in the image by the corresponding pixel in the template and taking the summation of the results. Thus, if gray-scale images and templates are used instead of a binarized images and templates, correlation can still be performed in accordance with the present invention.

In one embodiment, a threshold value between 0% and 100% is selected to determine an acceptable match ('thresh middle'). If the match is not acceptable, different portions of the input candidate center region are selected and additional

correlations are performed. These other portions can be rotationally and/or positionally shifted with respect to each other within the input candidate center region. In one embodiment, rotation steps of between 2 degrees and 5 degrees were found sufficient to achieve acceptable matching values. Thus, the input candidate image could be rotated ± 180 degrees or more with respect to the center region of the recognition template. In another embodiment, the results of each correlation is used to determine the selection of the next portion of the input candidate center region to correlate with the center region of the recognition template until a maximum match value for the recognition template center region is identified. The center of that portion of the input candidate center region deemed an acceptable match is selected as the best match reference point 1330.

The center region correlation procedure according to one embodiment of the present invention is discussed below with respect to three scenarios, A, B, and C:

Successive input candidate center region portions within the $X+m$ pixel by $X+m$ pixel area are correlated with the recognition template center region until all the desired portions have been correlated. The desired portions can be rotations and or position shifts relative to the candidate's reference point.

A: If no match is found, 'm' is increased in size, that is, the input candidate center region is enlarged, and additional correlations are performed with the recognition template's center region. If still no match is found, the user is then rejected.

B: If only one input candidate center region portion is successfully matched, that is, has a match value higher than thresh middle, that portion is selected as the best match

center region 1350 and the center of this region is selected as the best match reference point 1330.

C: If more than one input candidate center region portion exceeds thresh middle, one of these portions has a significantly higher match value, that portion is selected as the best match center region 1350. However, if several portions have approximately the same match value, each of these portions are selected as best match center regions for subsequent use in this matching procedure 1200.

Outlying Region Correlation 1240: Once one or more best match center regions 1350 are selected, each best match center region can be used as the basis for the outlying region correlations. For each best match center region 1350, the entire input candidate binarized image is rotated to correspond to the rotation for that best match center region. Then, location information 1340 for each of the outlying regions stored in the recognition template is used to locate a respective input candidate outlying region 1360 on the input candidate image. The size of each input candidate outlying region 1360, in one embodiment, is selected to be a square of $X+z$ pixels by $X+z$ pixels, where z is selected be less than m . Then, a similar correlation procedure is performed with respect to the procedure used for the center region correlation, except that the desired portions of each input candidate outlying region 1360 selected are not permitted to vary in rotation or position shift from one to another as much as the center region.

Various match parameters can be set by a system manager. For example, the threshold value for an acceptable match value for the center region and/or an outlying region, the number of outlying regions to correlate, and/or the number of outlying regions achieving an acceptable match value to accept a fingerprint as a match, can be set directly by the system

manager. The system manager can also indirectly set these match parameters indirectly by selecting a desired security level, for example, between 1 and 10. For example, in one embodiment, if two outlying regions fail to match, the user is rejected 1250, even if the center region matched.

Depending on the security needs of a particular installation, the number of outlying regions stored in the recognition template can be selected at enrollment. Thus, for example, the recognition template for access to a high security building can include ten outlying regions, whereas for a low security building, perhaps only three outlying regions need be stored in the recognition template.

Acceptance 1260: The user is accepted, that is matched, if the requirements for the selected matching parameters have been satisfied. In one embodiment of the present invention, all but one of the outlying regions compared must match, and a sufficient number, for example, between 3 and 10, of the outlying regions must have been available for correlation. An outlying region may not be available if the input candidate image is of a low quality.

One concern of using bitmaps for fingerprint matching is that if an unauthorized party somehow obtains the stored fingerprint image information, duplicates of the fingerprint, or images thereof, could be reconstructed. However, with the present invention, such reconstruction is impossible because the complete fingerprint bitmap is not stored in the recognition template. Instead, only selected regions of the fingerprint image are stored. Further, in one embodiment of the present invention, the location of these outlying regions, that is, the location information is encoded and/or encrypted.

Identifying an input candidate fingerprint from a database requires more time than verification simply because the input candidate image has to be compared to a larger number of stored images. The way the database is searched is critical to

reducing this time. When searching through a database of considerable size some kind of classification system reduces the number of stored fingerprints which have to be compared. Problems in classifying fingerprint images are well known. For example, the traditional classes and subclasses are not fully separable, that is some fingerprints can belong to more than one class. Also, some fingerprints do not fit into any of the classes, for example, fingerprints with scars, defective fingerprints, etc. Additionally, a hierarchical classification is useless unless the classification is 100 percent accurate and no such automated classification scheme is known to exist. Also, in a traditional hierarchical 'tree structure' database, the candidate image is first classified as being a specific type of fingerprint. Further subclassifications are then used within each class to locate a stored matching image. However, with such a hierarchical system, once a class or subclass is selected, further classification is only performed down that 'branch' of the hierarchical tree structure.

Instead, according to one embodiment of the present invention, a non-hierarchical database is used. In this database, certain characteristics of a fingerprint image, for example, the number of ridges crossing a certain line and/or ridge thickness, are numerically represented which permits each fingerprint to be located at a specific location in an N-dimensional space. When later searching in the database the input candidate image is represented using the same numerical representations and the N-dimensional fingerprint space is searched starting from the location of the input candidate fingerprint image in the N-dimensional space. Thus, the search is not limited to a certain branch of a database tree structure. Instead, the entire database can be searched, albeit starting at an initial point in the N-dimensional space from which the likelihood is high that the matching recognition template will be found nearby (quickly). Also, the portion of the database that is searched can be selected depending on

different factors such as time, false acceptance ratio (FAR), and/or false rejection ratio (FRR).

Each of the selected image characteristics corresponds to a coordinate of a recognition template's point in N-dimensional space. These coordinates can be stored in the recognition template along with the rest of the recognition template information. When the database is accessed for identification purposes, the coordinates for the input candidate image are calculated and some form of geometric distance between these coordinates and the recognition template coordinates stored in the database is calculated. The distance can be calculated, for example, by using the Pythagorean theorem or N-dimensional octagons (which require less computation). The first recognition template which should be attempted to be matched is the recognition template having coordinates located the shortest distance from those of the input candidate image.

As a specific example, according to one embodiment of the present invention, the N-dimensional space is defined with four dimensions. The template of this embodiment includes a square center region from the binarized image of 100 pixels by 100 pixels centered on the reference point. Along the four lines defining the outer perimeter of the square, the number of transitions from black to white and white to black are counted. These values provide four numbers as coordinates to define a point in 4-dimensional space. The coordinates for this point are stored along with the rest of the information forming the recognition template. When searching the fingerprint space, in this case 4-dimensional space, the distances between points in the 4-dimensional space can be calculated using straight forward Euclidian distances, that is, $d^2 = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2 + (v_i - v_j)^2}$.

APPENDIX A

Reference numbers "[]" correspond to the references at the end of this Appendix A.

5 Fingerprint recognition has mainly been associated with law-enforcement applications until recently. The idea of using fingerprints, or any biometric data, for identifying people in civilian environments has been regarded as science fiction. The fast, and persistent, technological evolution of microprocessors, memories, cameras (CCDs) during the last 20
10 years has made it possible to create Automated Fingerprint Identification Systems, AFIS, that are fast, and cheap, enough to be applicable for identification purposes in high-security areas. What has happened in the last few year is that the required components have become low-cost enough to implement
15 this technology in virtually any application requiring identification. Since the industry is very new there is not any one system that is dominant and not any one identification/verification approach that is regarded as the most appropriate.

20 This master thesis is an overview, and evaluation, of existing approaches including their trouble areas. The common way is to extract some points in a fingerprint that are considered unique for that finger and compare that data to an enrolled finger based on the distance between these points.
25 This is called feature extraction. We also describe some new methods to the identification/verification problem with a different approach: bitmap matching. This includes pre-processing, some bitmap analysis and matching. Finally some results are presented.

30 Introduction - The first use of fingerprints for identification purposes was done by the Egyptians to identify criminals and by the early Chinese to record business transactions, both around 3000 bc, see [1]. In modern days fingerprints started to attract attention in the

law-enforcement community during the 19th century and has since then been used extensively and increasingly all over the world.

The idea of using fingerprints as a mean of identification is based on their uniqueness and their consistency over time. No two persons are known to have identical fingerprints even though fingerprint patterns are inheritable and prints from monozygotic twins are highly alike in the large patterns, but not at all so with regard to small features, referred to as minutiae, see [2]. Furthermore, fingerprints do not change their characteristics over time, they stay identical from the time when they are formed in the womb throughout a persons entire life unless they are subject to deep cuts.

The ordinary way of collecting fingerprints has been to wet a person's finger in ink and roll the finger over a piece of paper. Today one uses almost exclusively optical capturing devices of different kinds, e.g. scanners and CCDs. Future capturing devices include CMOS-based readers measuring the capacitance between the finger's ridges/valleys and the active surface - the benefits being that they are cheaper and much more compact because no optics or lenses are needed.

The large fingerprint bases are stored digitally today as opposed to the fingerprint charts of earlier days, mainly because of easy access and the use of automated verification/identification algorithms. All approaches described in this thesis are based on optical capturing, subsequent A/D (analogous to digital) conversion and digital storage of the prints. This enables enhancement, restoration, binarisation and more complex image processing such, as FFT and vectorisation.

Up until recently identification through fingerprints has been used almost exclusively in criminal justice. The reason for this has been that the identification process has been very time-consuming and has required a fingerprint expert to manually search through large databases. Typically hundreds of

thousands of prints exists in the FBI fingerprint database. During recent years a few civilian applications has emerged, e.g access-control to high security areas such as banks (San Polo Bank, Italy) and industries (J2R Technology, France),
5 immigration control (Air Police Bureau, Taiwan), ATM machines (Standard Bank of Johannesburg, South Africa) etc.

Future applications seem at a first glance to be virtually limitless, anyplace where identification/personalised access is needed, the limitation being the price, accuracy and size of
10 the unit. Anything from police arms (the gun is useless unless the right person's finger is pressed against the butt) to bicycle-locks (you use your finger to open the lock instead of a key) might be future installations. More realistic in a near future are secure financial transactions (smart-cards,
15 internet) on a large scale basis, computer log-on (instead of or in addition to passwords and/or username) and access control (banks, research areas, military areas etc.). A few possible civil applications are: Access control; Time/Attendance systems; Smart-card verification; In addition to / instead of
20 PIN codes; Security boxes; Alarm systems; Software file access; Secure file transfer over computer networks; etc.

In biometrics, the science of identification of a person by their human characteristics, there is a need to measure the functionality of a system. This can be used in comparison
25 between different manufacturers and between different biometrics approaches, for example a retina identification system versus a fingerprint identification system. The most common biometrics are: fingerprint, retina, voice, hand, face and signature.

30 A. Properties of characterisation used are: (1) The accuracy of the biometric identification; (2) False acceptance rate (FAR), that is, the percentage of unauthorised persons accepted in error; and (3) False rejection rate (FRR), that is,

the percentage of authorised persons who are incorrectly denied acceptance.

These performance criteria are not easily interpreted, because there are several factors that can influence them.

5 First, there is a relationship between these two performance criteria. Often there exists some parameter that can be adjusted in order to decrease the FRR and increase FAR or vice versa.

FAR/FRR figures can be determined with 'one-try' or 'three-try' test protocols. In a 'one-try' test protocol, people are given only one chance to test their biometric data. After that, they are accepted or rejected. In a 'three-try' test protocol they have up to three chances before they are really rejected. If consecutive measurements are statistically
15 independent this improves the false rejection rate without really deteriorating the false acceptance rate. However, for some applications the three tries test protocol may not be acceptable.

In some biometric identification methods the bulk of the false rejections is caused by only a very small group who have very unstable biometric data. If it would be acceptable to
20 exclude this group and identify them in a different way, the false rejection rate could be improved dramatically.

The False Acceptance Rate is usually being tested by choosing random biometric data from a large population and trying to identify them with some selected data. However, when it is somehow possible to choose non-random biometric data that resembles the selected data, then the False Acceptance Rate could be much higher for this selected subset of biometric
25 data. This can be a problem when the biometric data is prone to easy visual inspection for resemblance such as faces or hand geometry, while this looking for resemblance is much more difficult for fingerprints or retina patterns.
30

The false rejection rate depends for a great deal on the experience of the users. If the users are trained and regular users of the biometric identification system, then the false rejection rate is much better than if they use it only occasionally.

B. Vulnerability to fraud - What kinds of possibilities are there for fraud, and what can be done about it? Here we must distinguish the possibilities for fraud when the person is willing to co-operate and to let his (biometric) data be duplicated, or when this data must be obtained in a secret way without the person knowing it.

C. Ease of use - Is the method of identification easy to use? Do we need special instructions to use it? Is it social acceptable, or does it frighten the general public?

D. Applicability - Is the method of identification applicable to everyone or is there a group of people who cannot use this method?

E. Speed of verification - How long does it take to identify a person with this method? Identification time excludes time needed to read the data from a card, or controlling an electronic gate door.

F. Size of storage for identification tokens - How many bytes do we need to store the data to identify a person? This is important if the data is stored in special media such as barcodes, magnetic cards or smart cards.

G. Long term stability - How stable are biometric properties after long periods of time? This indicates whether the method is useful for people who use it only once in a few years.

H. Proven technology - How often is the system used in "real-world" systems and how well does it perform? Unfortunately there can be no definitive conclusions, because of the on-going development of new techniques that require reconsideration. However, currently there are only a few

candidates that can offer a real system with adequate performance.

For absolute security the retinal reader is the safest solution. This could be acceptable for high-security access control systems, but probably not for systems meant for a more general public such as border control or automated financial transactions. For border control fingerprint identification and hand geometry are both viable and proven solutions.

The false rejection rate for the hand geometry is a little bit better than the false rejection for fingerprint identification. With respect to false acceptance the fingerprint identification is superior.

It is often argued that false acceptance is less important, because for a potential intruder it makes no difference if he has a chance of 1 in 10000 or 1 in 100 of getting through. In both cases he will probably be discouraged and think twice before attempting to intrude. But if fingerprint systems gets a general breakthrough and is widely used the intruder probably thinks over even one more time about what all that access control could mean.

Fingerprint classification - Historically, fingerprints have been divided into four, sometimes five, main structures; Loop (Left and Right), Whorl, Arch and Tent (Figs. 16 to 20) according to the 'Henry Classification Scheme'. The structures names comes from the pattern that ridges, the upper layer of skin, creates (The lower layer is called valleys). You can, with some imagination, see for example a tent or a whorl in a fingerprint. This system was created by Sir Edward Henry during the late 19th century, while situated in India, to prevent his workers from getting multiple wages by identifying them. The classification systems in use today by police agencies are based on the Henry System, with the addition of 3 subclasses, them being; Central Pocket, Double Loop and Accidental Whorl. A certain subset of all fingerprints is not

possible to classify according to this scheme, one example of this is shown in Fig. 21.

Minutia Points - Apart from different types of prints there are small micro-features, minutiae such as ridge endings, bifurcations etc. (Fig. 23)

There are mainly 7 different kinds of minutiae, schematically shown in Fig. 23. Typically, a fingerprint covering the top joint of a finger contains 50 to 150 minutiae points, the large span due to variable ridge width and individual differences. If 12 to 16 of these, from two separate prints, are matched exactly according to their relative location it is regarded as a match, i.e. the two prints originate from the same finger. The number needed to secure an identification might seem low, but one has to be able to identify partial prints, low quality prints (bad contrast, broken ridges due to dry fingers, over/under saturated prints), prints with small scars due to ordinary work etc.

The conventional way of performing identification is to first classify the print into one of the five previously mentioned main structures in order to minimise the portion of the database that has to be searched. The second step is to first locate a few characteristics as reference points, relative to which other characteristics are marked. If these reference points correspond to characteristics on the print with which it is to be compared more characteristics are located. The final decision is based upon how many of, and how well, these characteristics coincide. For a more thorough description of this verification method please refer to [3].

Nearly all Automated Fingerprint Identification Systems (AFIS) today uses some variant of the above explained method of locating minutiae. This is called 'feature extraction'. Since this has been the main approach for a fair amount of time we wanted to investigate the possibility of performing the verification process without using 'feature extraction'. The

main idea behind this is that the combination of a video camera, a CCD and a micro-processor - all optimised for this specific problem - could perform more complex image analysis during a very limited time-span.

5 Problem definition - The idea behind this master thesis is to look into the different approaches to the problem of correctly verifying a person's identity, or identify a person, using his/her fingerprint.

10 The focus is on defining the problem areas in the image analysis process and the available solutions to these problems. It is also our intention to try some new ideas such as bitmap matching, and some new methods to existing problems such as:

1. Defining a reference point.
2. Performing a vector representation of a fingerprint.

15 We have not made any full system measurements. This is partly because the routines have not been robust enough, but also because of the lack of any objective way of comparing two different systems.

20 Equipment - The equipment used was based on a 200-MHz Pentium computer with 64 MB of RAM. The 8-bit gray-scale images were captured by a TouchViewII fingerprint scanner from Identix, see Fig. 24, using the Intellicam framegrabber. The scanner uses the phenomenon called Total Internal Reflection, implying that only parts of an object that touches the surface
25 will be visual. This will be explained in the next section. Some testing was done on Veridicom's fingerprint sensor, illustrated in Fig. 25, which produces an image of the fingerprint based on difference in capacitance at a ridge or at a valley. The functions and algorithms were all written in
30 Matlab and C.

Fingerprint scanner - Schematically, this apparatus is shown in Fig. 26. When the ridges of the fingerprint are in touch with the glass surface, the light rays hitting that area are subject to total internal reflection. The rays hitting

areas where there are valleys, i.e. the finger is not in touch with the glass surface, are not reflected but transmitted and diffracted. The rays that are subject to total internal reflection hit the CCD and the signal from this is a digital representation of the image.

Image Analysis - This chapter contains the different processes of the image processing and matching. We have divided these into five sections: Binarisation; Reconstruction, rotation; Vector representation; Finding a reference point; and Fingerprint matching.

Binarisation - The information in a fingerprint is mainly found in the shape and not in the colour distribution. Therefore, one often binarises the images, i.e. convert it to an image that only consists of two colours, black and white. A big advantage that is achieved is the reduction of the amount of data (8 times assuming an 8-bit greyscale image). The visualisation to the human eye is also improved which is a pleasant side-effect when working with the images. The binarisation is in its simplest form carried out with the use of thresholding. Every pixel value that exceeds a threshold will be set to the maximum value. All other pixels will be set to zero.

There are two main types of thresholding:

Global thresholding - A thresholding value is chosen and is applied over the entire image. Median value and mean value (Equation 1) are widely used. In the fingerprint application it is not a well-suited alternative because some areas are darker or lighter depending on the pressure of the finger. An example of this is shown in Figs. 27 to 29.

$$t = \frac{1}{(m*n)} \sum_{i=1}^m \sum_{j=1}^n A(i, j) \quad (1)$$

Local thresholding - An extension of the above idea is to divide the image into smaller areas and then apply a threshold to each area. This will improve the quality of the processed image but will cause edges between the areas. This procedure is known as local thresholding. The smallest possible area, that is still functional, is a moving (3 x 3) pixel neighbourhood. That means that only the mean- or median value of the closest pixels will affect the thresholding of the centred pixel. This method has the disadvantage of high sensitivity to noise that will cause many unwanted islands (a white pixel surrounded by black pixels or the opposite) in the image.

A thresholding method using a ridge-valley orientation detector has been developed by Stock and Swonger [6]. That is the direction of a 'line', formed by upper and lower skin layer, is approximated. The routine detects, at each pixel location of the fingerprint image, the local orientation of the ridges and valleys of the finger surface, and produces an array of regional averages of these orientations. Eight slitsums, S_i , are calculated for this purpose, all represents a direction. A slitsum is calculated for the pixel C by taking the mean of the pixels marked in Fig. 30 with its direction number. To calculate for example the vertical direction, one takes the mean of the pixel positions marked with 1. A zero means that the pixel should be ignored. A ridge consists mainly of black pixels and therefore a slit sum that is parallel with it will be small. The binariser then sets the output pixel to white if the value of the central pixel, C, exceeds the average of the pixels of all slits, that is, if Equation 2 is valid.

$$C > \left(\frac{1}{32} \right) \sum_{i=1}^8 S_i \quad (2)$$

For the binarised image A (0 representing black and 1 representing white) in Fig. 31 slitsum(1) for pixel (6,6) is

calculated as: $A(4,6)+A(2,6)+A(8,6)+A(10,6) = 2$. In the same way, all sums becomes (2, 0, 1, 2, 3, 3, 4, 3). The average is calculated to 0.56 which is less then $A(6,6)$. The pixel is therefore set to black. From this result it is pretty obvious that $A(6,6)$ should be a pixel in a ridge that somehow got distorted. Of course this is done in a greyscale image but the general idea is better showed in a binarised image.

The slit comparison formula that sets the output pixel to white if the average of the minimum and maximum slit sum exceeds the average of all the slit sums is shown in Equation 3, below.

$$\frac{1}{2} (s_{\min} + s_{\max}) > \frac{1}{8} \sum_{i=8}^8 s_i \quad (3)$$

The motivation for this formula is as follows. If a pixel is in a valley, then one of its eight slits will lie along the (light) valley and have a high sum. The other seven slits will each cross several ridges and valleys and these slits will therefore have roughly equal, lower sums. The average of the two extreme slit sums will exceed the average of all eight slit sums and the pixel will be binarised correctly to white. Similarly, the formula causes a pixel lying on a ridge to be binarised correctly to black. This formula uses the slits to detect long structures (ridges and valleys), rather than merely using their constituent pixels as a sampling of local pixels as Equation 2 does. It is able to ignore small ridge gaps and valley blockages, since it concerns itself only with entire slits and not with the value of the central pixel.

Stock and Swonger found that they obtained good binarisation results by using the following compromise formula, rather than using either Equation 2 or Equation 3 alone: the output pixel is set to white if

$$4C + s_{\min} + s_{\max} > \frac{3}{8} \sum_{i=1}^8 s_i \quad (4)$$

5 This is simply a weighted average of Equation 2 and Equation 3, with the first getting twice as much weight as the second.

10 We found out that Equation 4 could be improved by extending the orientation detector to have longer slits. This is a better variant because of the differences in the ridge widths will affect the results. The extended orientation detector causes much better thresholding if it is wider ridges, while 'normal' ridges will remain the same. Perhaps an even further extension could be made, but its computational cost will be less than the improvement achieved. Each frame of pixels added slows the process down because of the extra 16 positions that must be checked.

15 As described above we conclude that general thresholding methods are not good enough for fingerprint images. The common structures of a print must be taken under consideration to obtain images suited for our approach.

20 Reconstruction, Rotation - As seen in Fig. 35, there are quite a few white pixels in the ridges, which obviously are wrong. Most of these can be eliminated by using the morphological operation called Majority. Majority sets a pixel to white if five or more pixels in its (3x3) neighbourhood are white. Otherwise it is set to black.

$$B(i, j) = 1, \text{ if } \sum_{k=1}^{i-1} \sum_{l=1}^{i-1} A(k, l) \geq 5 \quad (5)$$

25 By repeating this procedure, an almost noise free image is created when the pixel values don't change anymore. Fig. 36

shows the result when a majority operation is performed on the fingerprint in Fig. 35 six times.

In identification systems that use feature-extraction, the next step is to find the skeleton of the fingerprint, see [7].
5 There are several methods to do this, but the main idea is to remove pixels on the boundaries of the pattern without allowing objects to break apart. Fig. 37 shows the skeleton of Fig. 36 with some of its features marked.

Referring to Figs. 38 and 39, sometimes, it can be very
10 difficult to decide whether for example a ridge-ending is a true one or if it is a result of poor image quality and the reconstruction process. Some criteria like the maximum number of pixels that can be missing in a ridge and still not be considered a ridge-ending must be applied, and further
15 reconstruction must be done. When this is done, however, it is a trivial task to extract the features: all black pixels that just have one black neighbour must be a ridge-ending and all black pixels that have more than two black neighbours must be a ridge-bifurcation. In the enrollment procedure these are
20 usually registered by their location and the angle of the connected ridge. This is the big advantage of feature extraction; the small amount of data that is needed to be stored.

In pattern matching there is a need to rotate the images
25 an arbitrary angle. This is important because it is very hard to place the finger on the scanner at the same angle that it was enrolled. Even with a video screen present, which very seldom is the case in a real-world application, it is nearly impossible to distinguish between a few degrees difference.
30 The image rotation matrix, T , with rotate angle α is defined according to:

$$T = \begin{pmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{pmatrix} \quad (6)$$

The new coordinates for the image, when rotated radians are given by:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} \cos\alpha & -\sin\alpha \\ \sin\alpha & \cos\alpha \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \quad (7)$$

This, however, leaves us with an image where the gray-levels of some pixels become undefined. A better approach to this is to go the other way around: calculate for each pixel in the new, rotated image, which pixel in the non-rotated image that should correspond to it.

$$\begin{pmatrix} x \\ y \end{pmatrix} = \frac{1}{(\cos\alpha\cos\alpha + \sin\alpha\sin\alpha)} \begin{pmatrix} \cos\alpha & \sin\alpha \\ -\sin\alpha & \cos\alpha \end{pmatrix} \begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} \cos\alpha & \sin\alpha \\ -\sin\alpha & \cos\alpha \end{pmatrix} \begin{pmatrix} x' \\ y' \end{pmatrix} \quad (8)$$

This is an interpolation problem. We choose a method called nearest-neighbour, it simply states that the pixel with the shortest distance from the calculated position will do.

Vector representation - The idea of vectorisation is to find the predominant directions in the image, either for every pixel or for larger areas (where the pixel-directions are averaged over the area). If this is done for large areas it is a way of compressing the amount of data in the image when looking for larger structures. As for fingerprint analysis this is typically done when trying to classify fingerprints or for locating the 'core point' and / or 'delta points', c.f. [4] and [5], which are defined as in Fig. 40. A 'core point' is defined as the topmost point on the innermost upward recurving ridge, where a recurving ridge means that it curves back on itself. A 'delta point' is defined as the point of bifurcation, in a delta-like region, on a ridge splitting into two branches which extend to encompass the complete pattern area.

There are several ways of finding the predominant direction for a pixel. We will describe two different ways of performing this where one method is highly mathematical and requires relatively high computational capacity, whereas the other is very straight-forward and easily implemented. They first one is referred to as the 'Gaussian derivative filter' and the other one is referred to as the 'Minimum absolute difference'. For a Gaussian derivative filter, one creates a qualitative gaussian surface, i.e:

$$z = e^{-(x^2+y^2)/\sigma} \quad (9)$$

where x and y are the co-ordinates of the filter relative (0,0) defined as the innermost point of the filter matrix. One example of this is shown in Fig. 41.

$$x \in \{-10,10\} \quad y \in \{-10,10\} \quad \text{and} \quad \sigma = 50 \quad (10)$$

This function is then derived with regards to x and y respectively

$$\begin{aligned} \frac{dz}{dy} &= \left(-2 \frac{y}{\sigma}\right) e^{-(x^2+y^2)/\sigma} \\ \frac{dz}{dx} &= \left(-2 \frac{x}{\sigma}\right) e^{-(x^2+y^2)/\sigma} \end{aligned} \quad (11)$$

Different combinations of these two are then combined to form directional filters for an arbitrary angle, ϕ , as shown in the example in Fig. 42.

$$z(\phi) = \cos(\phi) \frac{dz}{dx} + \sin(\phi) \frac{dz}{dy} \quad (12)$$

For $\phi = \pi/8$, the above equation becomes:

$$z\left(\frac{\pi}{8}\right) = \cos\left(\frac{\pi}{8}\right) \frac{dz}{dx} + \sin\left(\frac{\pi}{8}\right) \frac{dz}{dx} \quad (13)$$

5 The whole image is then filtered with as many filters as the number of desired directions. Subsequently, every pixel attains a value, for every direction, corresponding to how well its direction coincides with the filter's dito. One then finds which direction is predominant. Directions for larger areas are then calculated based on the average over the area. What should be mentioned here is that if one wants to define directions for areas the size of 8*8 pixels the averaging should be done over slightly larger areas, e.g. 12*12 pixels, whereas a smoothing is brought about, (see Fig. 43 and Fig. 15 44).

 The vectorised image in Fig. 43 was created using a derived gaussian filter. Every direction (8*8 pixels) represents the average direction over 8*8 pixels. The word average is not all correct since there is no average for 20 directions, i.e. the average of 10° and 350° being 0° and not 180°. Even though it is possible to find the correct average for two directions (using "mod") operators) and subsequently grouping the directions 2-2 and then 4-4 and so on. Instead, what is done is that one lets opposite directions cancel each 25 other until all are confined to the same 180 degrees interval after which the mean is calculated.

 The vectorised image in Fig. 44 was created using a derived gaussian filter. Every direction (8*8 pixels) represents the average direction over 16*16 pixels as compared 30 to 8*8 pixels above.

 Minimum absolute difference - The Minimum absolute difference method is based on the fact that the pixel values along a ridge/valley varies less than across the same ridge/valley, see Figs. 45 and 46.

If the directions are seen as the slits with the same number in Fig. 47, ranging from 1 (vertical, 90 degrees to 8 (-67.5 degrees) rotating clockwise. The centre pixel, C is the pixel for which one is trying to define a direction. The directions, 1 to 8 are then defined as the slits in Fig. 47.

One then performs the summation below:

$$S(dir) = \sum_{i=1}^6 C\text{-pixel}_i \quad (14)$$

The summation is done for $i=1:6$ because all directions are represented six times in the example shown in Fig. 47, this value could just as well be lower or larger - it would only mean that the summation is done over a smaller or larger area. The value S is calculated for all eight directions. One then finds the lowest value of S and this direction is the predominant for the particular pixel C.

Our idea was to somehow use the vectorised image to define a 'reference point'. This has been done before ([4] and [5]) but with the limitation that a 'core point' can not be defined for fingerprints of the type 'Arch'. Since we weren't limited to finding the exact 'core point' we wanted to thoroughly investigate if it could be possible to find a reference point, which could be virtually any point in the inner area of the print.

It is very hard to judge the quality of a vectorised image. At the same time as one wants a smooth image one does not want to average out too much information. All settings of the parameters below have been done by observing the vectorised images and how well these describe the original image, as anyone can conclude the optimal value depends on what kind of information one is looking for and the quality of the incoming image.

A very low-quality image would need more averaging whereas in high-quality images one would look for finer details (i.e. larger changes in the vectorised image).

We started out using the gaussian derivative filter because it gives a good vectorised image even though it is a very slow method (ca 60 seconds for 374*388 pixels on a Pentium 200MHz using Matlab for the matrix operations and customised c-routines for all loops). The different parameters that can be set for the Gaussian filter, are the size of the filter (i.e. how large the matrix z should be) and the sigma-value (i.e. how steep the inclination/slope should be).

The optimal size of the filter is strongly correlated to the resolution of the incoming digitised image. The filter should be large as to comprise one valley and one ridge. Since not all people have the same ridge-width one has to compromise and select a filter somewhere in between. On a resolution of 500 dpi (appr. 20 lines/mm) we concluded after various tries that a filter size of 7*7 pixels was the optimal.

As for the sigma-value it is not as obvious how this is related to the characteristics of the image. One could say that a small sigma-value should be used if the incoming image has a very good contrast and vice-versa.

For images with good contrast on a 8-bit scale we used the value 0.8. This value was concluded using trial and error and ocular observation of the vectorised images.

Minimum absolute difference - This method is much more straight-forward. The computation time is under 100 ms for a 388*374 pixel image (all written in C and executed on a 200MHz Pentium).

The most significant parameter for this method is how many frames, i.e. what size of the matrix in Fig. 47, one should use. The matrix should cover one ridge and the neighbouring valleys, or vice-versa. The optimal number of frames is then correlated to the resolution and the actual thickness of the

structures, in mm. As mentioned above, not all people have the same ridge-width. The optimal size, for a 500 dpi image, is a matrix the size of 13*13 pixels, as shown in Fig. 47.

Furthermore, one can set thresholds for how large the variance of the different slit-sums for pixels must be. This gives an indication of how well the direction is defined. This can then be used as a weight when averaging over larger amount of directions.

The only way to say whether a vectorisation is good or not is to make a subjective judgement of how well it corresponds to the original image. Our conclusion on this matter is that the vector images using the 'Minimum Absolute Difference' are almost as good a representations as those using the 'Gaussian Derivative Filter'. The differences are hard to find and even harder to evaluate. The examples shown in Figs. 48-50 are not taken to show the qualitative difference but to demonstrate how similar the vector representations are, and the difficulty in saying which one is the most accurate.

As for our work the time aspect by far out-weights the possibly less accurate representation. Because of this we decided to proceed with the routine based on 'Minimum Absolute Difference'.

Reference point - The idea of a reference point is not new to the field of fingerprint analysis. In the classical, i.e. manual, way of matching prints all minutia are given positions relative the core point of the print. A core point is defined as the topmost point on the innermost upward recurving ridge, where a recurving ridge means that it curves back on itself (see Fig. 51).

The main difference between this 'core point' and the point we are trying to establish is that there are no restrictions on where in the print the 'reference point' is located or on how the local structures are in the vicinity of the same. The only feature that is important is that this

point is the same for every sample of the same print. (this of course means that it should not be situated too far out on the edges of the print, since this area might not be present in another sample of the print).

5 The information in a fingerprint is both global (larger structures as the different classes) and local (minutia). This implies a problem since one can not search locally for a reference point since many minutia are 'the same' when looked at separately. Furthermore one can not search too globally since what could exclusively define a reference point is found locally.

15 The only places where we could find any mention of how a 'reference point' should be located were in abstracts of Japanese patents, so this seems like an area not very well researched by the scientific community.

20 Our approach could be divided into three different approaches, where two are based on the same principle whereas the third is an altogether different approach. These methods are:
FFT; Ridge Count; and Searching a vector image.

25 The first two are thoroughly explained in this paper but the third is left unexplained due to the fact that it might be regarded for a patent. When judging the success of any of these routines we have set that if none of the reference points differ more than 30 pixels from that of any other sample it is a successful method for that print.

30 First a 1-dimensional Fourier transform is applied to every row and every column of the image. The spectra for these are then slightly averaged. Finally one looks at the point where there are the most high-frequency components. The point where the highest vertical frequency coincides with the highest horizontal frequency is defined as the 'reference point' of the fingerprint.

These frequencies are found by doing a FFT on the rows and the columns of the images. The frequency spectra are then sent through a band-pass filter taking away the frequencies that obviously do not correspond to any of the structures in the print. This could be very low frequencies due to variations in the background light of the input image, or very high frequencies due to internal structures of the single ridges / valleys. What should be mentioned is that the magnitude of the different frequency components are not linear to the actual occurrence of those frequencies when using an FFT. The FFT still gives pretty good images of the main frequency components and it can, to some degree at least, be used to find the dominant frequency component.

Fig. 52 shows a frequency analysis of a fingerprint. The plots show the main frequency versus the row (y) and column (x) respectively. The arrow point towards the point in the print defined as the reference the routine based on FFT. The two lowest plots show the need for averaging before one defines what row and column has the highest main frequency.

Fig. 53 shows the same analysis for another print. This print has much more of a circular structure and it is also cut so that only relevant information - meaning the area where there actually is a print - is considered when the frequency analysis is done. What can be seen is that the averaging is crucial in this print as well - look at the plot titled 'x-led no filtering' and the frequency components around the 100th column.

What we found rather quickly was that the 'reference point' is not very well defined using FFT, i.e. there is not a smooth structure with a single, well defined top. This was verified when we used this method on several samples of the same print. If the method is to be useful it has to define the same point in the print for all samples of this print. In Fig. 54, 9 samples of the same print are shown and the arrows

indicate the reference points defined by the FFT routine. As is easily seen from this figure that the reference point is not unambiguously defined, e.g, the reference point in sample no. 9 (left to right, top to bottom) would be 50 pixels to the right of the reference point in sample no. 7.

The parameters which are correlated to the output are as follows:

Band-pass filter - What should be mentioned is that it is not the image that is filtered but rather frequency spectra of the rows and columns. This is done to screen out certain frequencies that might otherwise dominate the subsequent analysis. The band-pass we used was, for horizontal frequencies, 'number of columns'/16 to 'number of columns'/2 and the equivalent for vertical frequencies. An averaging was done in this routine as well with the following mask:

```
[0.25  0.33  0.5  0.67  1  1  1  0.67  0.5  0.33
0.25]
```

in order to average out differences between neighbouring frequencies.

Smoothing - When the dominant frequencies for all rows and columns are calculated (see the two bottom plots in Fig. 52 and Fig. 53) these are averaged using a filter-mask specified below (which results in top left, centre left and centre right plots of Fig. 52 and 53):

```
[0.27  0.4  0.8  1  0.8  0.4  0.27]
```

The settings of both these parameters were optimised by inspecting the results from the routine and trying to judge improvement and deterioration. Since there is no objective way of doing this it is all based on 'how it looks' but that is image analysis at its best.

Extensive tests and optimisation of the different parameters of the routine did not improve the result in any major way. As an approximate figure this method succeeds with about 50 percent of all prints.

Ridge Count - This method is similar to the FFT-method in that it looks at the number of ridges crossing a vertical/horizontal line. One then defines the reference point as the point where the vertical line and horizontal lines with the most ridge-crossings intersect. A certain set of rules are applied to the case when more than one line has the same number of ridge-crossings.

The incoming image is first binarised and restored. For every row and column one counts all shifts from black to white and vice versa. The row with the most shiftings is regarded as the row of the reference point. One then defines the point (row, column) with the highest ridge count as the starting point for the second part of this routine.

Using the row and the column values from the starting point one then finds the closest ridge on the left side and on the right side of the point (i.e. the first occurrence of a black pixel) see point A in Fig. 55. The final column is found by following this ridge, and the two neighbouring ridges, upwards until their top point and the median of these three columns values is the column used. This point (row, column) is then the reference point, see point B in Fig. 55.

By following a ridge we mean that one moves upwards in a black area with white boundaries. When one reaches a pixel location where there is a white pixel above and moving to the left or right, don't "opens any doors up" that is can't find any black pixels in a lower row, one have find the highest point of that ridge. Note that a higher ridge point means a lower row number). This method is somewhat shown in Fig. 55.

The result of applying this method on the 9 samples used in Fig. 54 is shown in Fig. 56. The result from applying this method on multiple samples of the same print were better than that of the FFT method. This method gave satisfactory results on most prints but could not handle all. There are certain fingerprint structures where this method 'jumps' between two

points depending on which parts of the outer areas of the print that are present. Since it has to yield the same point for virtually any sample of a print this is not good enough.

As an indication of the quality of the routine it can be said that it succeeds with around 90 percent of all prints.

Even though this method will not be explained it can be said something about it's performance. According to the tests done on this method the success rate is 99.7 percent. The biggest difference is that this method uses both global and local structure when defining a reference point.

Even though there is information to be found in a frequency analysis of a fingerprint it is not unambiguous enough to define a reference point. The reason for this is that these routines are based on the information in the whole print. This means that if, in two different samples, different parts are present apart from the area around the reference point these method will yield different answers for these prints. Since there is no way of defining which area is to be analysed before the reference point is located this dilemma is quite impossible to solve. The method based on the vector image on the other hand uses both global and local structure and is therefore more applicable.

Fingerprint Matching - Many different fingerprint matchers are known. The most prevalent of these compare search print (scanned) and file print (enrolled) minutiae, while others use direct optical comparison methods of images as wholes. Fingerprint matchers that compare search and file minutiae generally may be categorised using two discriminants: (1) Whether the matcher is spatial or topological; and (2) Whether the matcher uses information 'local' to minutiae exclusively. These discriminants form a framework for understanding the present fingerprint matcher in the context of those that are known methods. These types will now be explained in detail.

A purely spatial matcher uses the geometric relationships among minutiae of a print without regard to the identity of ridges which then terminate, bifurcate, or the form of the structure. The Cartesian co-ordinates of each feature are recorded along with an estimate of the angle that each feature exhibits with respect to the frame of the print image. For instance, the angle of a ridge-end can be estimated from a short segment of the dark pixels leading away from the minutia along the ridge.

A topological matcher uses the connectivity of ridges among minutiae to establish relationships among them in a single print. If two ridge ends terminate the same ridge, on which there is no other minutia, then there is a specific relationship between the minutiae, that may be exploited by the matcher. This relationship, of being connected by the ridge, holds irrespective of the distance between the minutiae. For instance, the number of ridges crossed by a line drawn from the core of the print to each minutia can be taken under consideration.

Not only is spatial and topological matchers are known, hybrid forms are known as well. A hybrid form of a matcher would use both geometric and ridge information about minutiae.

The second discriminant for categorising a fingerprint matcher is whether or not it uses information "local" to minutiae exclusively. If a spatial matcher compares the co-ordinates in the frame of the image of search minutiae to those of file minutiae, that matcher uses "global" information, not information strictly local to individual minutiae. The global information is the reference to the frame of the image.

A spatial matcher that uses only local information would construct and use relative geometric relationships within small groups of minutiae, such as pairs. The geometry would be referenced only to a local frame, not to that of the full

image, making the matcher less vulnerable to distortions of the skin than a matcher that uses global information.

We decided to try another method for the matching step. Common sense says that a image of a fingerprint that contains more information than just the minutiae must be a better description of the finger characteristics. With a bitmap, information like the thickness of the ridges, the curvature of the ridges and scars are registered among the minutiae locations. Naturally this method has its drawbacks. Spatial displacement, rotation and local distortion are problems that arise.

Local distortion can for example occur when the finger is placed on the scanner and then twisted. Parts of the image will then rotate a different number of degrees depending on the pressure and the distance to the centre of rotation. In this section we assume non-rotated and non-distorted images for the purpose of simplicity. Rotation is no actual problem as described earlier, except for eventual time demands and we believe that the distortion issue can be overcome with further work. Spatial displacement is of course also time consuming when a matching location of the bitmaps shall be found. So we will show the importance of finding a well-defined reference point as well as display the uniqueness of the matching position found.

Two bitmaps will be compared, one with the whole fingerprint present, and one smaller bitmap which is centred around the reference point. The bitmaps are binarised. This has a big advantage if an implementation in a microcomputer is done due to the logical properties of just two colours. A simple not(xor) operation is all that is needed.

In the surface plot in Fig. 59, bitmaps are compared, A (Fig. 58) and B (Fig. 57). B is matched in every position on A around its centre point. The height of the surface is a measurement of how good the matching score is. The size of B

can be chosen quite arbitrary but it is an advantage if not necessary to include some kind of specific structure. For example parallel ridges will not do so good because it occurs in most peoples fingerprints. The reference point finders described will very seldom find a point within such an area.

The matching score is calculated as the number of pixels that are of the same colour in the two bitmaps at a certain location divided by the area of the smaller image. A matching score of 1 is a perfect match and 0 is when no pixels at all match. The latter is however highly unlikely.

The maximum score is 0.81 and should be considered a good hit because of the fact that we neglected the rotation. Fig. 60 shows the best-matched position.

The maximum score of a match between image C shown in Fig. 61, and image A shown in Fig. 58, was 0.59. The difference of the true bitmap and the false bitmap is over 20 percent. This is an unusual bad match, which probably comes with the fact that two prints have different main structure, A is a loop and C is a whorl. False matching score can also be very low if it differs a lot in ridge thickness and therefore have an overall less amount of black pixels.

Even if two prints from two fingers are quite similar to the human eye, the maximum matching score very seldom exceeds 0.67. This seems to be some kind of a magic limit and we suspect that it can be shown with the help of a probability theorist. Some prints have reached over 70 percent but it is very rare.

The matching was done in Matlab and the smaller bitmap was matched at (60 * 60) positions. With a stable reference point this search area could have been diminished drastically. We also think we have shown that there really exists a distinct maximum if the prints are from the same finger, and that this method is worth further investigation.

As an example of the results Fig. 63 shows a three dimensional plot of the score for 8 prints matched against each-other. As can be seen the difference between a match (e.g. print 1 against print 1) and a non-match (print 1 against print 2) is quite clear. The z-axis is numbered from 0 to 1000 where 1000 is a perfect match. The x- and y-axis are not numbered according the prints but each quadrant represents one print versus another, a total of 64 quadrants. Every quadrant represents 30 matchings which means that the Fig. 63 shows a total of 1920 matches ($30 \times 8 \times 8$).

Overall it can be said that on basis on the results presented it seems that matching fingerprints using bitmap matching could be done with the restriction that the prints may not be distorted. As compared to existing systems using feature extraction, which also have restrictions on distortion, the advantage is that all information in the matching area is used.

Even though our algorithms are not yet stable enough, we believe that we are on a interesting track. The next step would be to integrate the routines into a full system where their robustness could be tested on larger amounts of prints.

We will continue to work on the project at our industrial sponsor, and perhaps integrate it into a larger system that verifies other biometric data as well as fingerprint data.

References -

1. D.K. Isenor and S.G. Zaky "Fingerprint Identification Using Graph Matching" Pattern Recognition, Vol. 19, No 2, pp 113-122, 1986] and [Henry C. Lee and R.E. Gaensslen "Advances in Fingerprint Technology" ISBN 0-8493-9513-5.
2. C.H. Lin , J.H. Liu , J.W. Ostenburg et al. "Fingerprint Comparision 1: Similarity of Fingerprints" Journal of Forensic Sciences, JFSCA, Vol. 27, No 2, April 1982, pp. 290-304.
3. Pp 48-53 in "Advances in Fingerprint Technology", H.C Lee and R.E. Gaensslen, ISBN 0-8493-9513-5.

4. V. S. Srinivasan and N. N. Murthy "Detection of singular points in fingerprint images" Pattern recognition. Vol 25 No 2, pp 139-153, 1992.

5. C. V. Kameswara and K. Balck "Finding the core point in a fingerprint" IEEE Transactions on computers, Vol c-27, No 1, January 1978

6. R. M. Stock and C. W. Swonger, "Development and evaluation of a reader of fingerprint minutiae", Cornell Aeronautical Laboratory, Tech. Report cal no xm-2478-x-1:13-17, 1969.

10 7. R.C Gonzales and R.E Woods, "Digital Image Processing", Addison-Wesley, 1992

15 Thus, it is apparent that in accordance with the present invention an apparatus and method that fully satisfies the objectives, aims, and advantages is set forth above. While the invention has been described in conjunction with specific embodiments and examples, it is evident that many alternatives, modifications, permutations, and variations will become apparent to those skilled in the art in the light of the foregoing description. Accordingly, it is intended that the present invention embrace all such alternatives, modifications and variations as fall within the scope of the appended claims.

20

What is claimed is:

1. A fingerprint processing method comprising the steps of:
obtaining an image of a fingerprint comprising ridges and valleys;

5 searching the image to locate a reference point; and
 selecting the reference point and a region in the vicinity
of the reference point as a recognition template for the image.

2. The method of claim 1, wherein the obtaining step further
comprises the steps of:

 applying the fingerprint to a scanning device;
 scanning the fingerprint to generate an image signal; and
 storing the image signal as a digital image.

15 3. The method of claim 2, wherein the searching step further
comprises the steps of:

 vectorizing the digital image;
 selecting a starting sub-area in the vectorized image;
 scanning from the starting sub-area along an orientation
20 of each subsequent sub-area to locate a first sub-area having a
horizontal orientation, the first sub-area included in a first
horizontal structure;

 scanning from the first sub-area across acceptable
structures and along a path of acceptable sub-areas until an
25 unacceptable sub-area is located; and

 selecting the center point of last acceptable sub-area as
the reference point.

30 4. The method of claim 2, wherein the selecting step further
comprises the steps of:

 calculating the geographic center of the digital image;
and
 selecting the geographic center as the reference point.

5. The method of claim 2, wherein the selecting step further comprises the steps of:

 binarizing the digital image;

 determining the row of the digital image which has the
5 greatest number of binary transitions;

 determining the column of the digital image which has the
greatest number of binary transitions; and

 selecting a point in the image by following a path
starting from a point in the image having the row and the
10 column as coordinates.

6. The method of claim 1, wherein the searching step comprises the steps of:

 selecting a starting point;

15 following along at least one ridge proximate the starting
to locate a ridge of a first type;

 selecting adjacent ridges of the first type along a
predetermined path to locate a ridge of a second type; and

 selecting a point on the last located ridge of the first
20 type as the reference point.

7. The method of claim 1, wherein the selecting step comprises the steps of:

25 selecting the region to include the reference point, the
region having a size and a shape; and

 storing the recognition template.

8. The method of claim 1, wherein the selecting step further comprises the steps of:

30 selecting other regions, each of the other regions having
a respective size and a respective shape, each such other

region located with respect to the reference point according to relative location information; and

selecting the other regions, and the respective relative location information for each respective other region as part
5 of the recognition template for the image.

9. The method of claim 8, further comprising the step of storing the recognition template.

10. The method of claim 8, wherein the storing step further comprises the step of:

encrypting one or more of the region, the other regions, and the relative location information.

11. The method of claim 9, wherein the storing step further comprising the step of:

compressing one or more of the region, the other regions, and the relative location information.

12. A fingerprint matching method comprising the steps of:

obtaining an image of a fingerprint comprising ridges and valleys;

searching the image to locate a reference point;

selecting the reference point and a region in the vicinity
25 of the reference point;

selecting at least one recognition template, each recognition template comprising a template reference point and a template region;

correlating at least a portion of the region with the template region to generate a correlation result; and
30

determining whether the correlation result exceeds a predetermined matching requirement.

13. The method of claim 12, further comprising the step of;
obtaining from the recognition template, relative location
information of at least one other template region;

5 selecting another region from the image utilizing the
relative location information with respect to the template
reference point;

correlating at least a portion of the another region with
the other template region to generate a correlation result; and

10 determining whether the correlation result exceeds a
predetermined matching requirement.

14. A fingerprint processing method comprising the steps of:

obtaining sequential multiple images of a fingerprint
comprising ridges and valleys;

15 determining dynamics of the obtaining step by comparing
the multiple images to each other.

15. A method according to claim 14 further comprising the step
of:

20 determining from the dynamics if the fingerprint is real.

16. A fingerprint information storage method comprising the
steps of:

25 obtaining from a fingerprint values for each of a number
of fingerprint characteristics;

assigning each of the values to a respective coordinate,
the coordinates defining a point in a dimensional space having
the number of dimensions; and

30 associating information concerning the fingerprint with
the point.

17. A method according to claim 16, further comprising the
step of:

THIS PAGE BLANK (USPTO)

locating information concerning other fingerprints based on the proximity of other points in the dimensional space associated with other respective fingerprints.

5 18. A fingerprint processing device comprising:

a sensor for detecting a fingerprint and for generating an image signal corresponding to the fingerprint;

a processor for receiving the image signal and for identifying a reference point and a region in the vicinity of the reference point in an image formed from the image signal; and

a storage device for storing information concerning the reference point and a portion of the image.

15 19. A fingerprint processing device according to claim 18, wherein the device further comprises:

a correlator for comparing information received from the storage device and information concerning the region in the vicinity of the reference point.

20

20. A storage template for a fingerprint processing system comprising:

a first region bitmap;

a reference point location;

25 outlying region bitmaps; and

relative location information, the relative location information corresponding to the location of each of the outlying region bitmaps with respect to the reference point location.

THIS PAGE BLANK (USPTO)

1/27

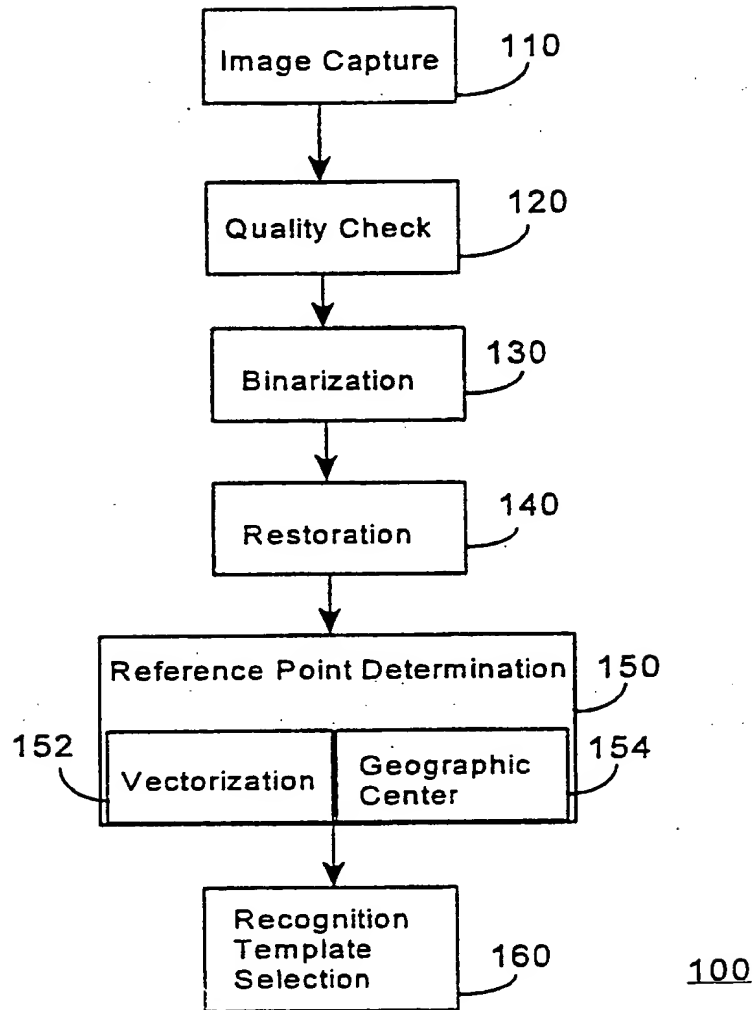


Fig. 1

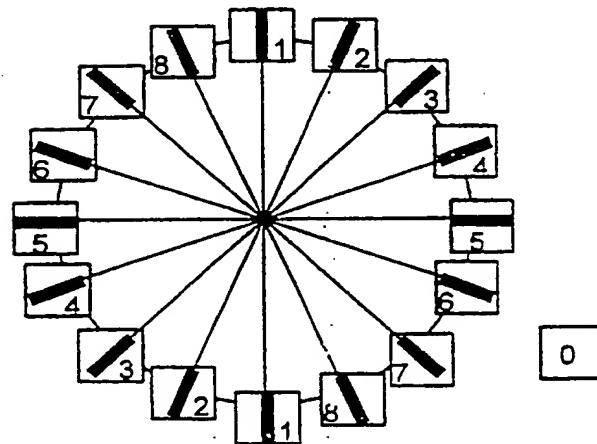
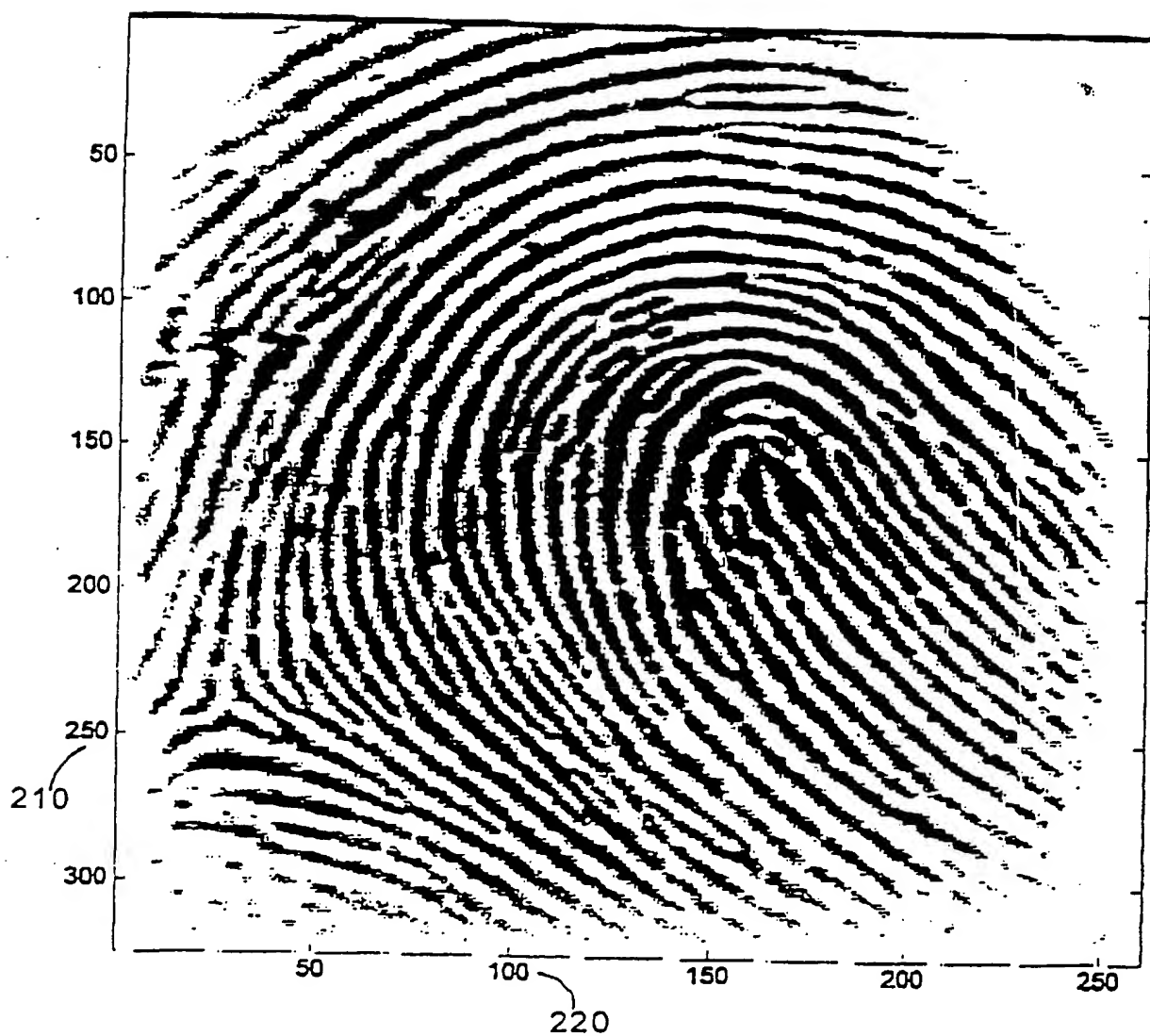


Fig. 4

THIS PAGE BLANK (USPTO)

2/27



200

Fig. 2

THIS PAGE BLANK (USPTO)

3/27

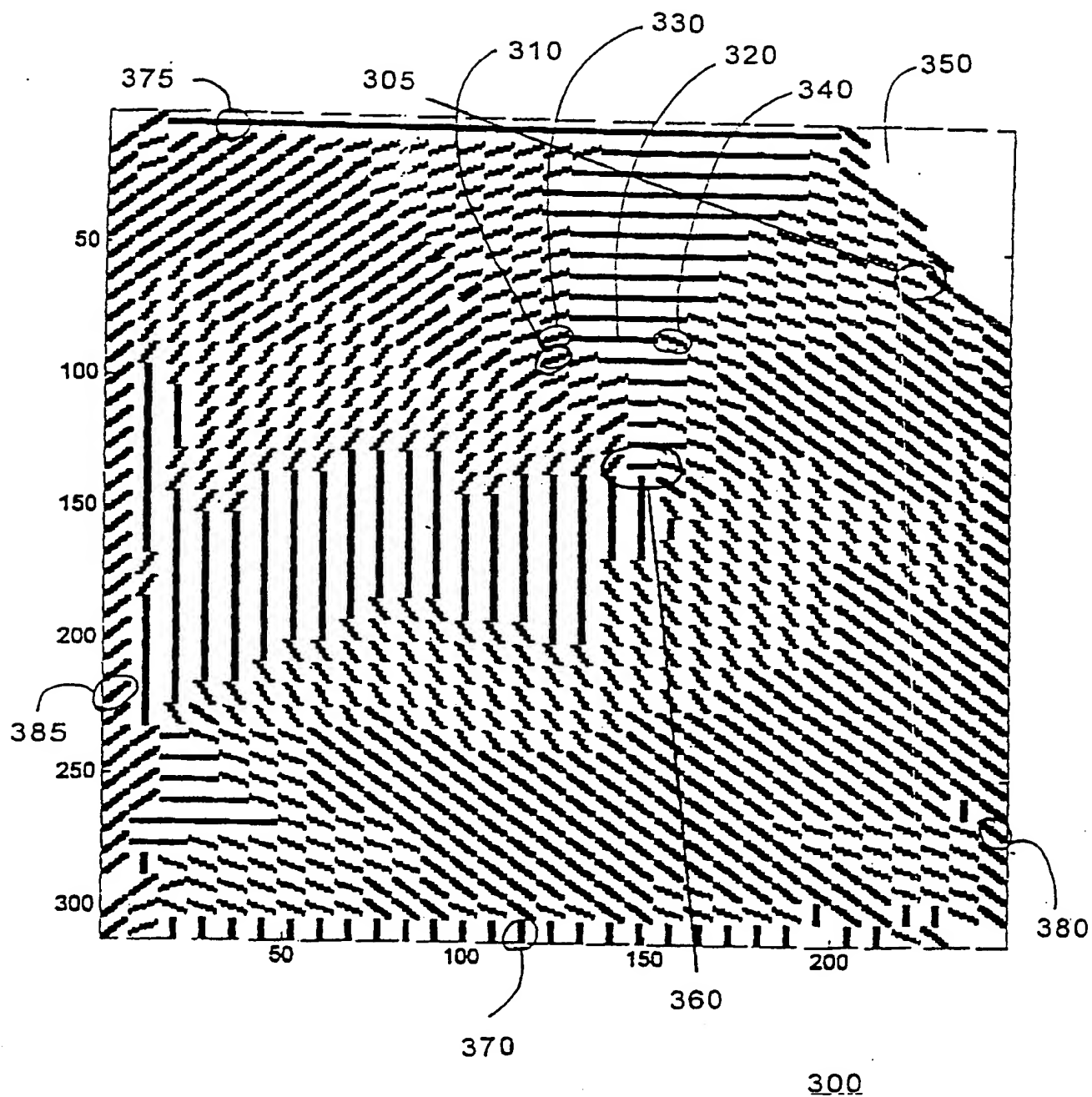


Fig. 3

4/27

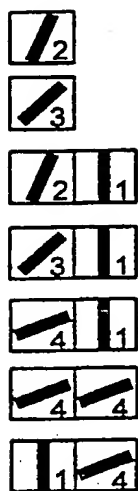


Fig. 7

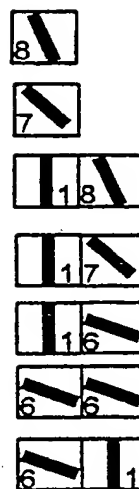


Fig. 8

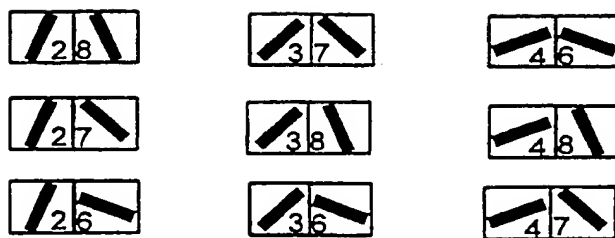


Fig. 5

THIS PAGE BLANK (USPTO)

5/27

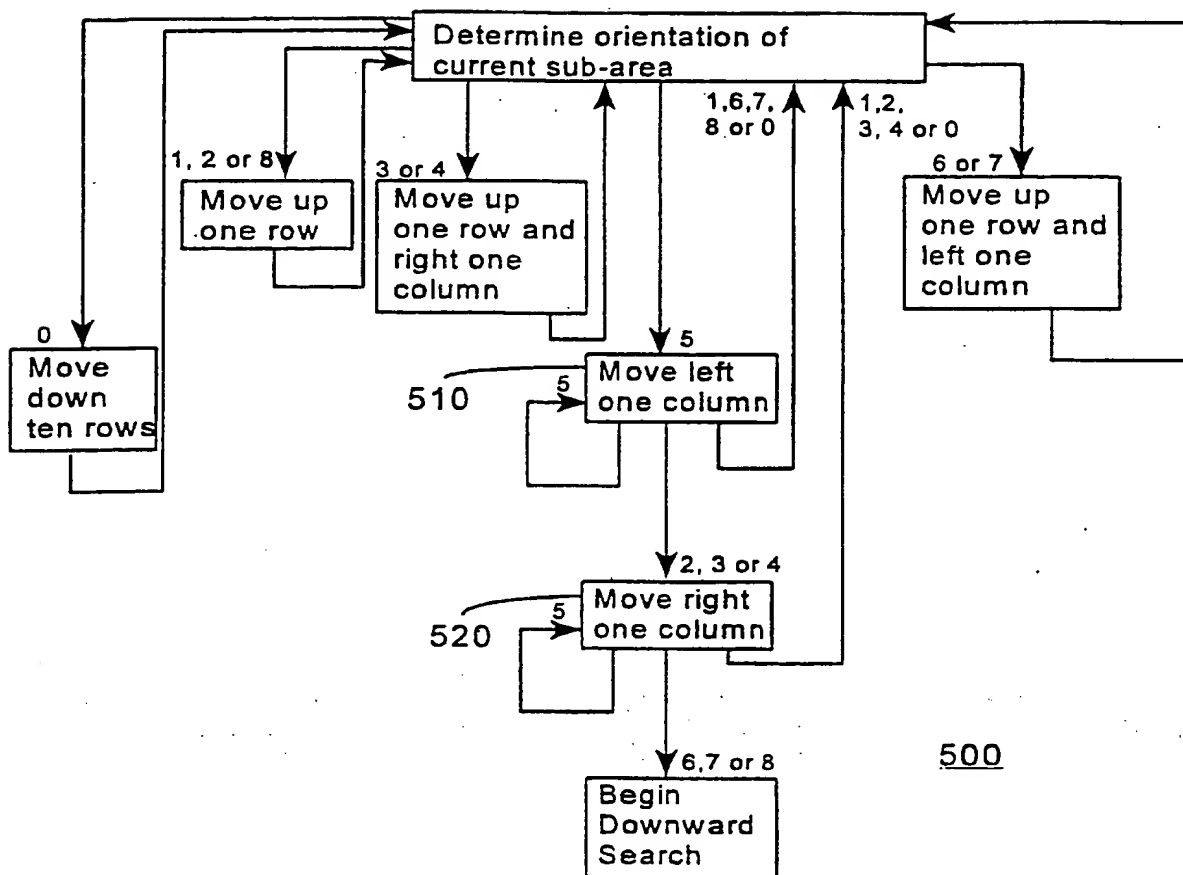


Fig. 9

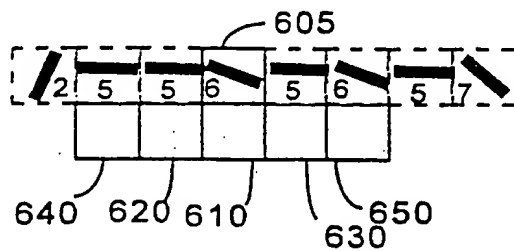


Fig. 6

THIS PAGE BLANK (USPTO)

6/27

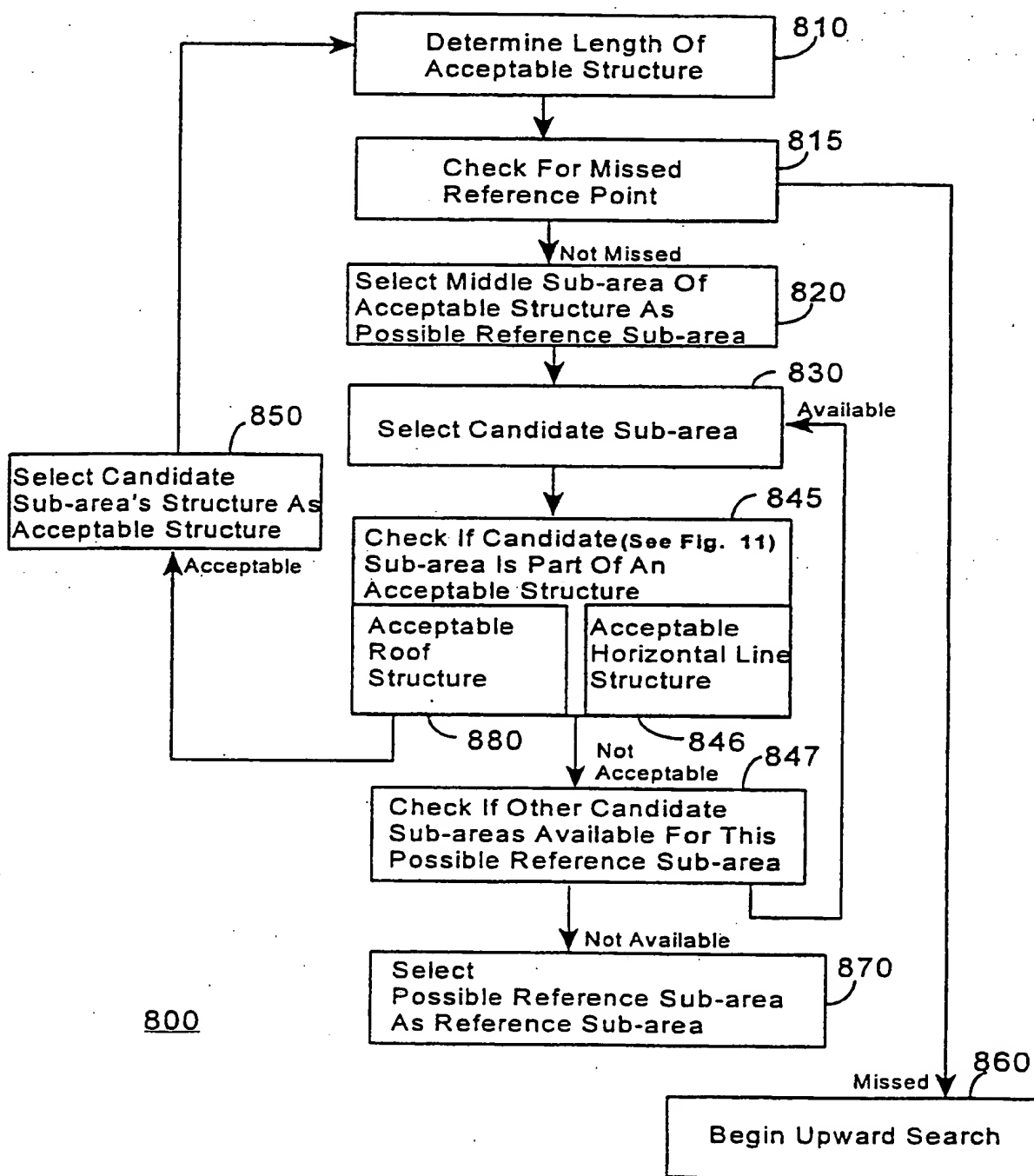


Fig. 10

THIS PAGE BLANK (USPTO)

7/27

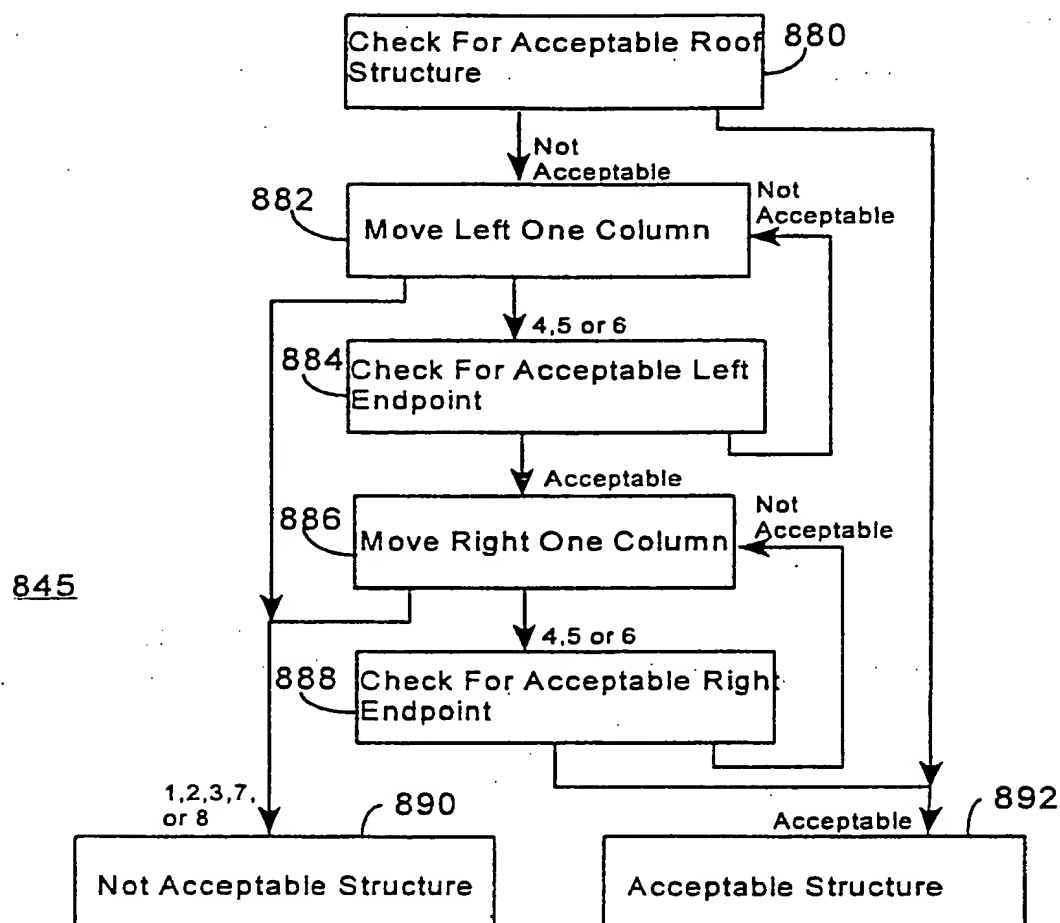


Fig. 11

THIS PAGE BLANK (USPTO)

8/27

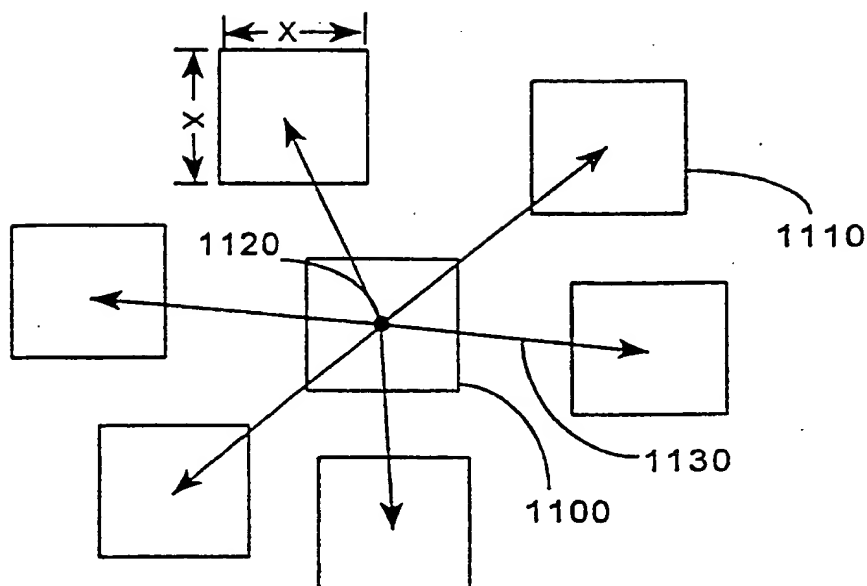


Fig. 12

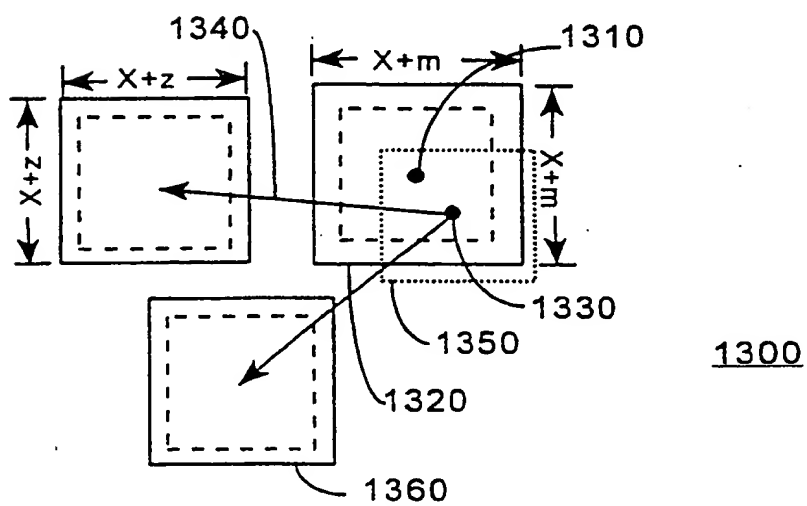


Fig. 14

THIS PAGE BLANK (USPTO)

9/27

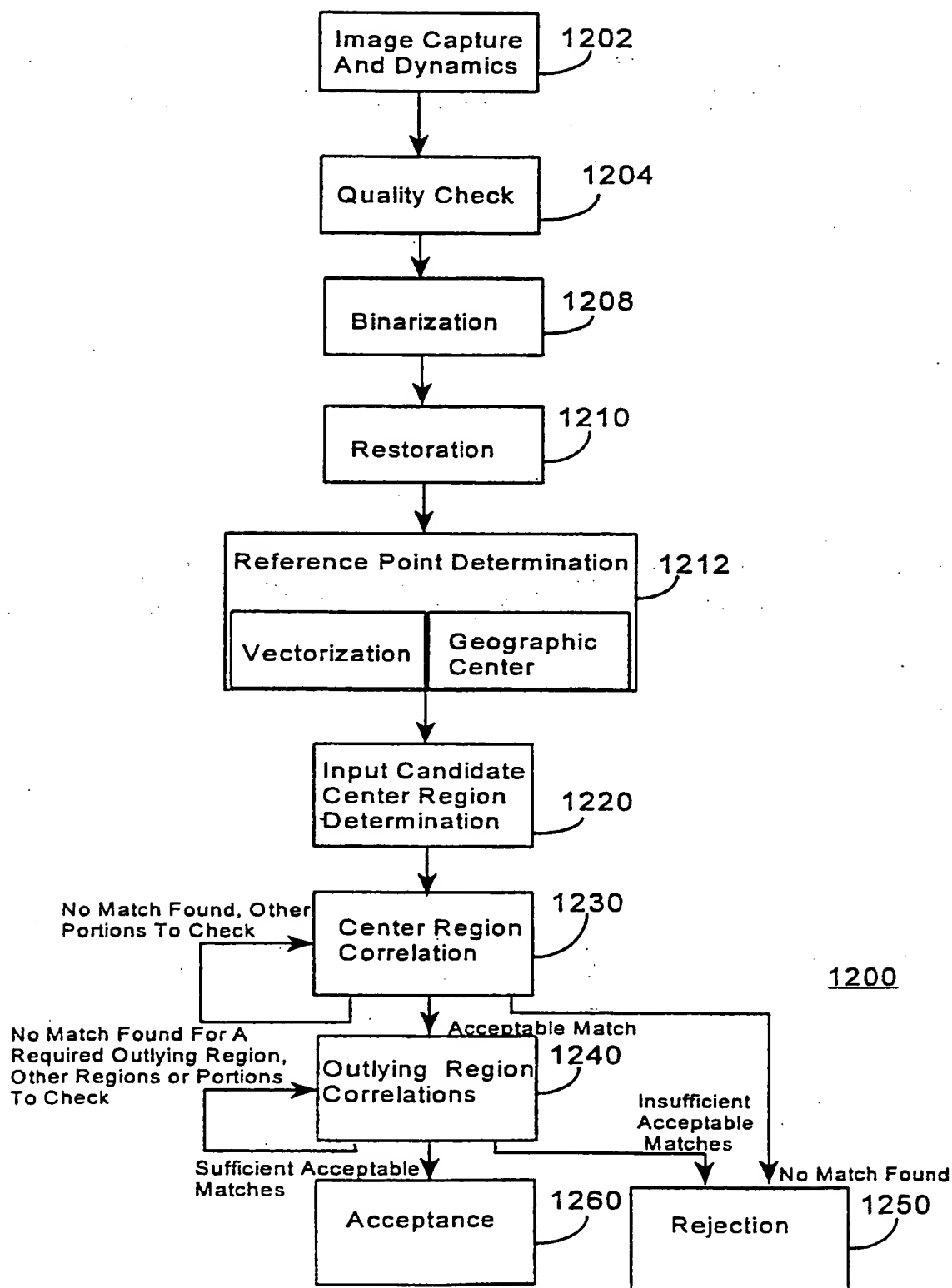


Fig. 13

THIS PAGE BLANK (USPTO)

10/27

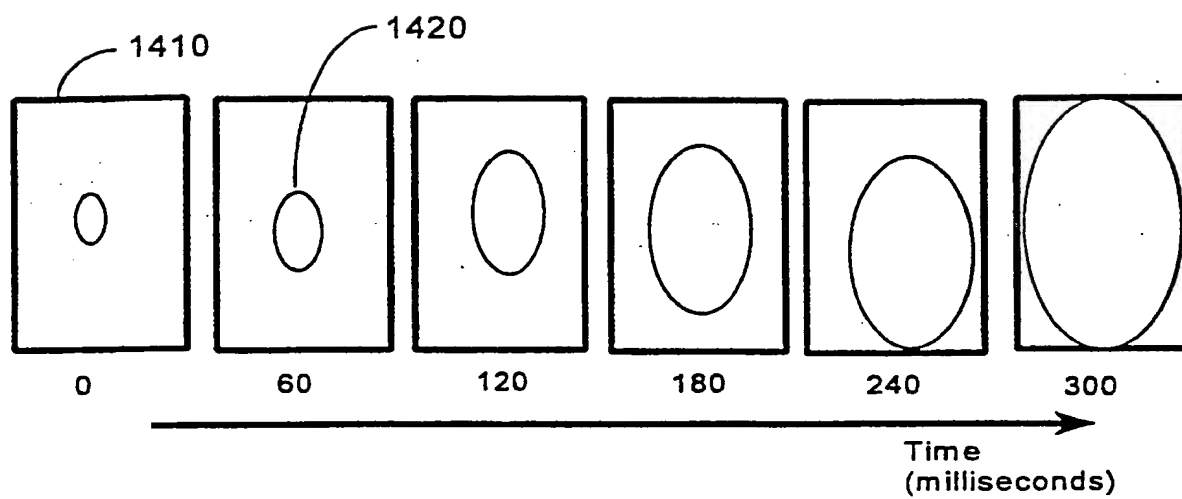


Fig. 15

THIS PAGE BLANK (USPTO,

11/27



Fig. 16



Fig. 17



Fig. 18



Fig. 19



Fig. 20



Fig. 21

THIS PAGE BLANK (USPTO)

12/27



Fig. 22

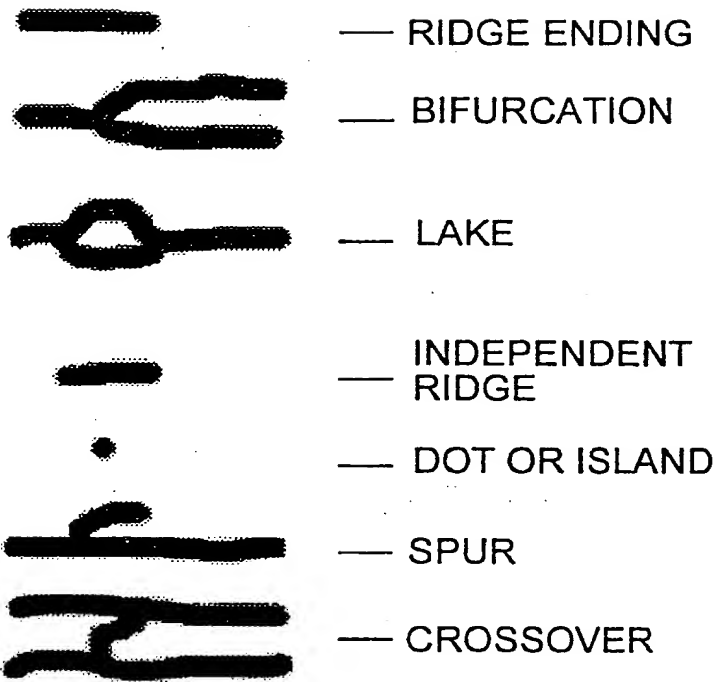


Fig. 23

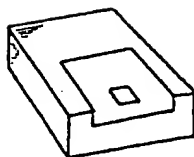


Fig. 24

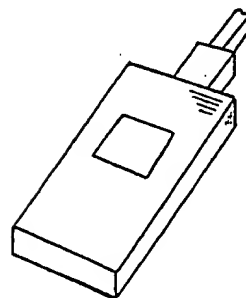


Fig. 25

THIS PAGE BLANK (USPTO)

13/27

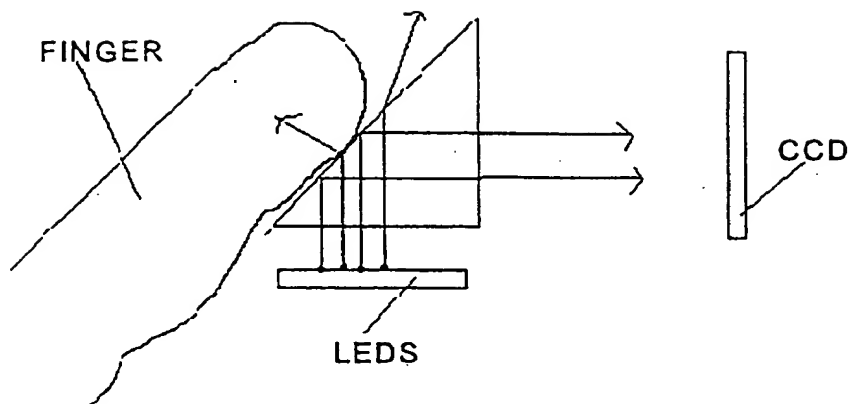


Fig. 26

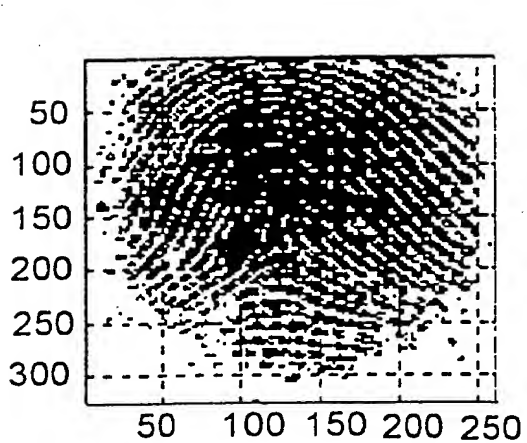


Fig. 27

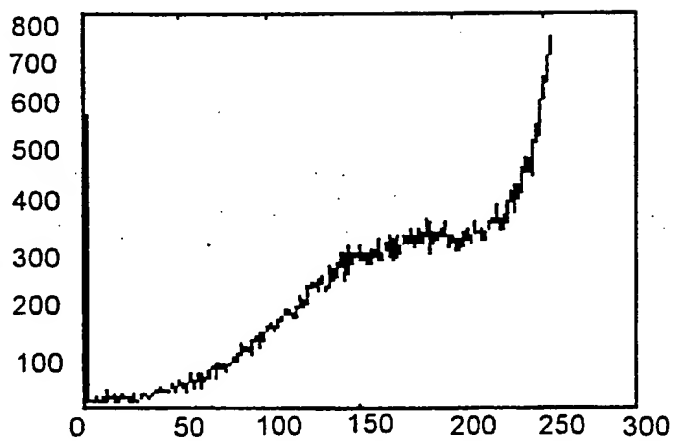


Fig. 28

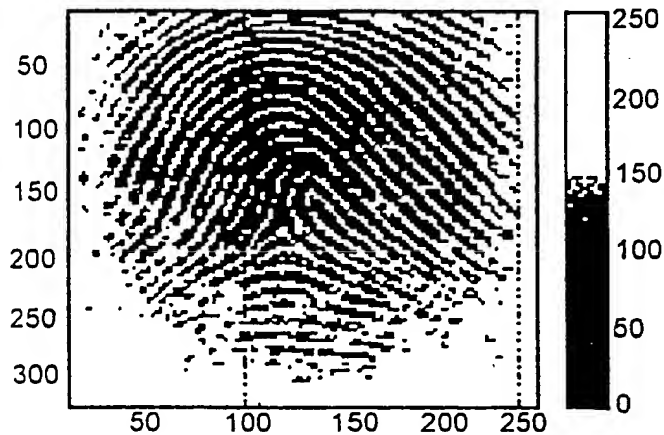


Fig. 29

THIS PAGE BLANK (USPTO)

7	0	8	0	1	0	2	0	3
0	0	0	0	0	0	0	0	0
6	0	7	8	1	2	3	0	4
0	0	6	0	0	0	4	0	0
5	0	5	0	0	0	5	0	5
0	0	4	0	0	0	6	0	0
4	0	3	2	1	8	7	0	6
0	0	0	0	0	0	0	0	0
3	0	2	0	1	0	8	0	7

Fig. 30

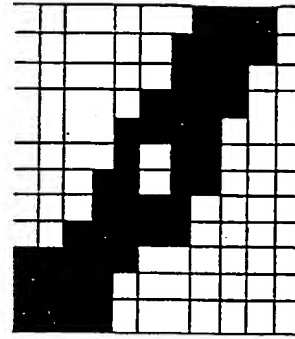


Fig. 31

7	0	0	8	0	0	1	0	0	2	0	0	3
0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	7	0	8	0	1	0	2	0	3	0	0
6	0	0	0	0	0	0	0	0	0	0	0	4
0	0	6	0	7	8	1	2	3	0	4	0	0
0	0	0	0	6	0	0	0	4	0	0	0	0
5	0	5	0	5	0	0	0	5	0	5	0	5
0	0	0	0	4	0	0	0	6	0	0	0	0
0	0	4	0	3	2	1	8	7	0	6	0	0
4	0	0	0	0	0	0	0	0	0	0	0	6
0	0	3	0	2	0	1	0	8	0	7	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	2	0	0	1	0	0	8	0	0	7

Fig. 32

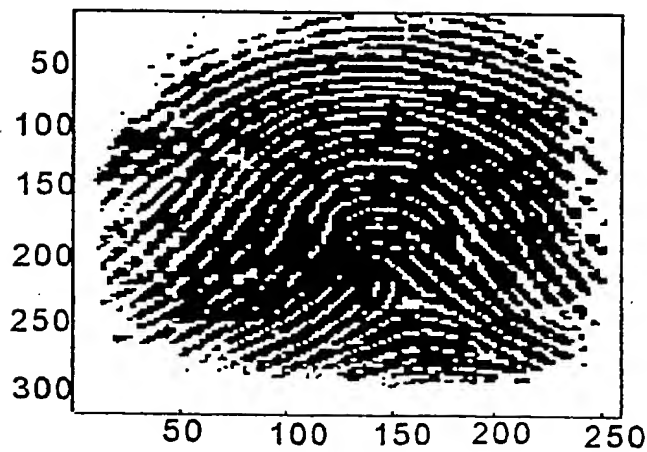


Fig. 33

THIS PAGE BLANK (USPTO)

15/27

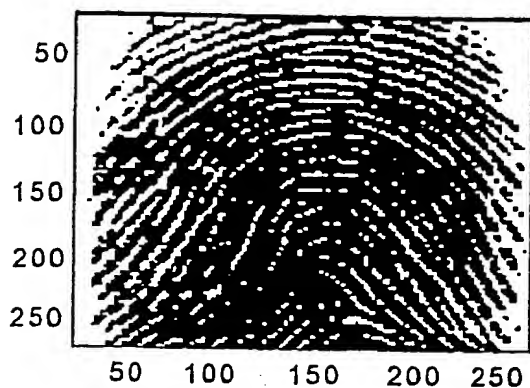


Fig. 34

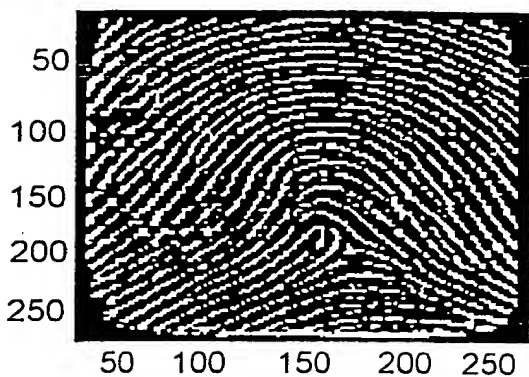


Fig. 35

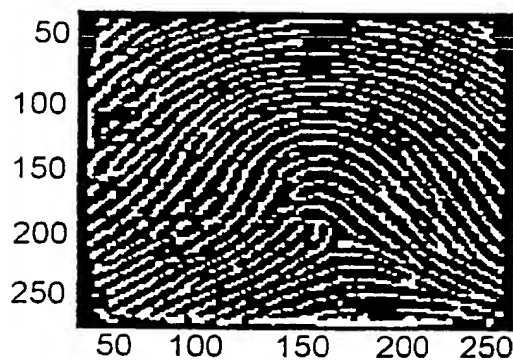


Fig. 36

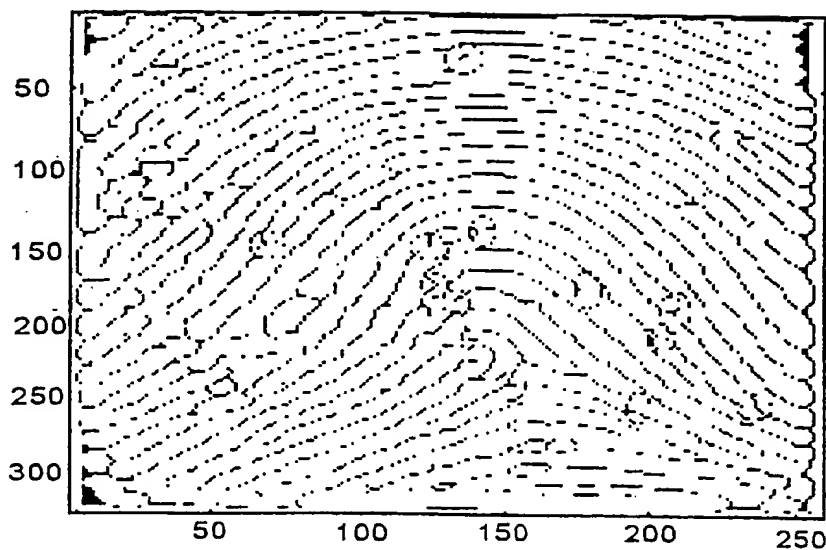


Fig. 37

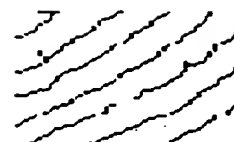


Fig. 38

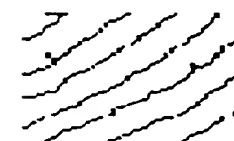


Fig. 39

THIS PAGE BLANK (USPTO)

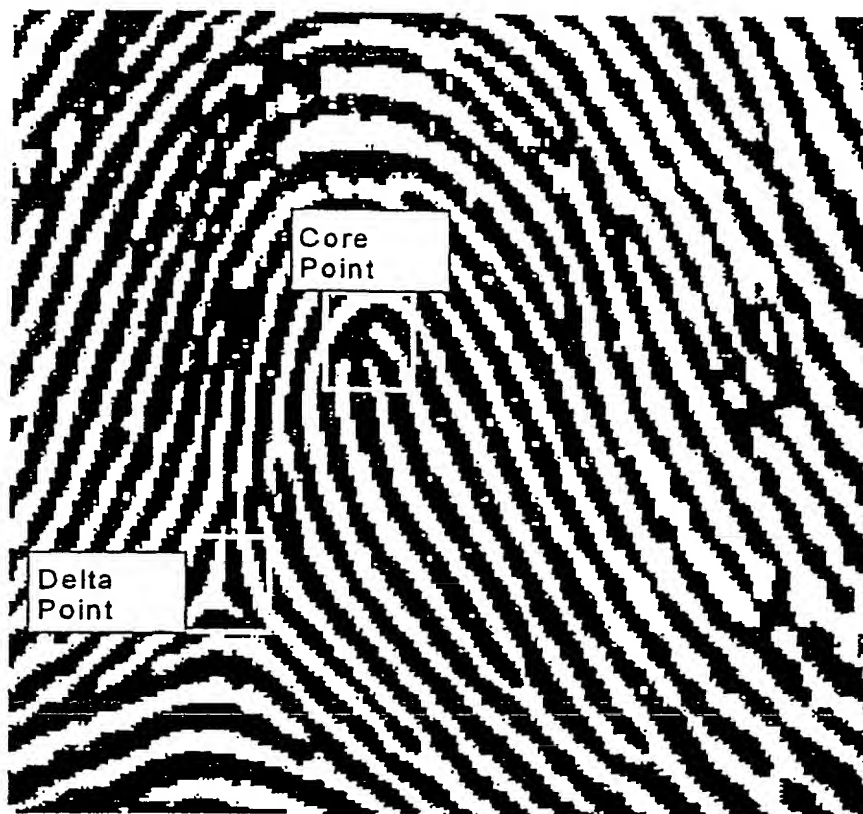


Fig. 40

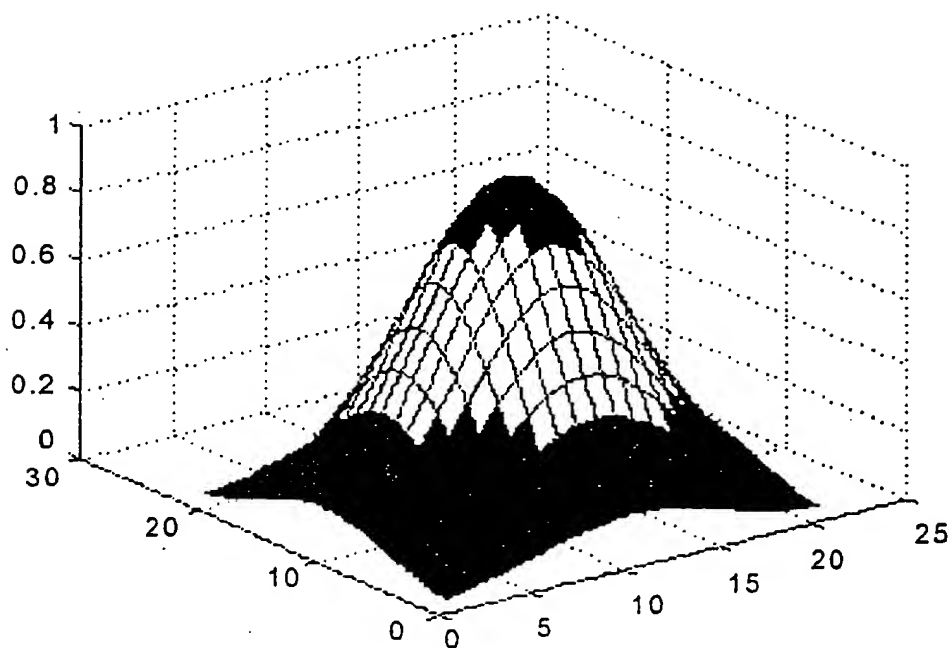


Fig. 41

THIS PAGE BLANK (USPTO)

17/27

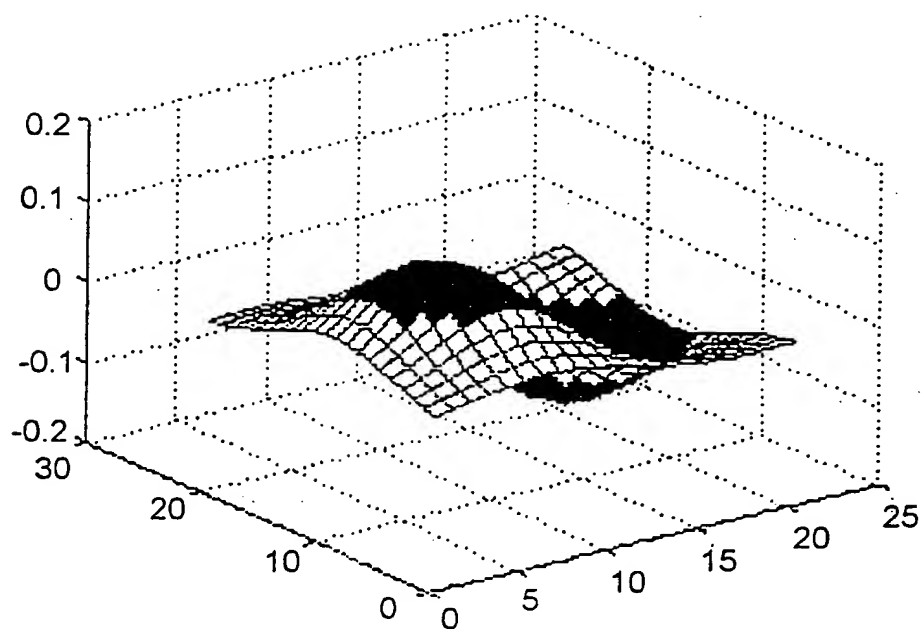


Fig. 42

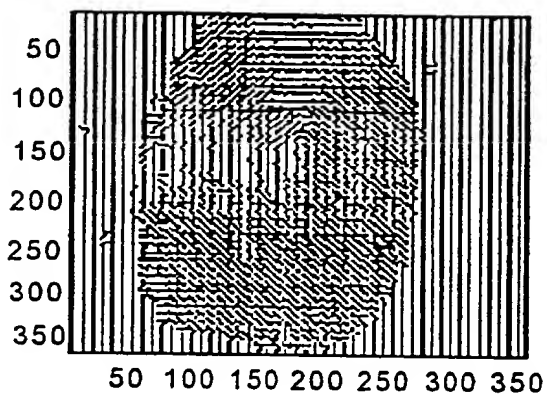


Fig. 43

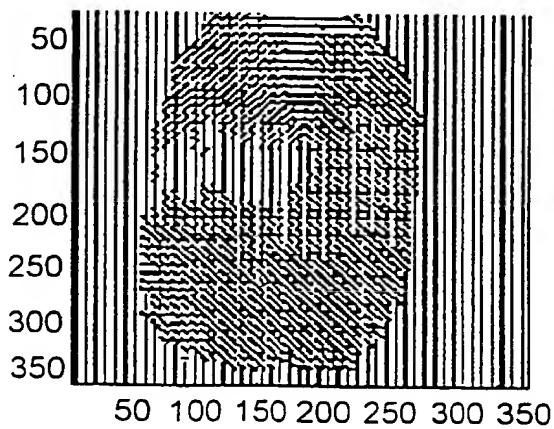


Fig. 44

THIS PAGE BLANK (USPTO)

18/27

$$A = \begin{bmatrix} 5 & 4 & 3 & 2 & 1 & 0 \\ 4 & 3 & 2 & 1 & 0 & 1 \\ 3 & 2 & 1 & 0 & 1 & 2 \\ 2 & 1 & 0 & 1 & 2 & 3 \\ 1 & 0 & 1 & 2 & 3 & 4 \\ 0 & 1 & 2 & 3 & 4 & 5 \end{bmatrix}$$

Fig. 45

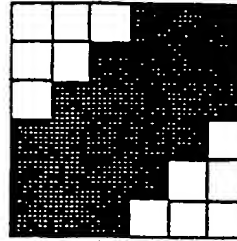


Fig. 46

7	0	0	8	0	0	1	0	0	2	0	0	3
0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	7	0	8	0	1	0	2	0	3	0	0
6	0	0	0	0	0	0	0	0	0	0	0	4
0	0	6	0	7	8	1	2	3	0	4	0	0
0	0	0	0	6	0	0	0	4	0	0	0	0
5	0	5	0	5	0	0	0	5	0	5	0	5
0	0	0	0	4	0	0	0	6	0	0	0	0
0	0	4	0	3	2	1	8	7	0	6	0	0
4	0	0	0	0	0	0	0	0	0	0	0	6
0	0	3	0	2	0	1	0	8	0	7	0	0
0	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	2	0	0	1	0	0	8	0	0	7

Fig. 47

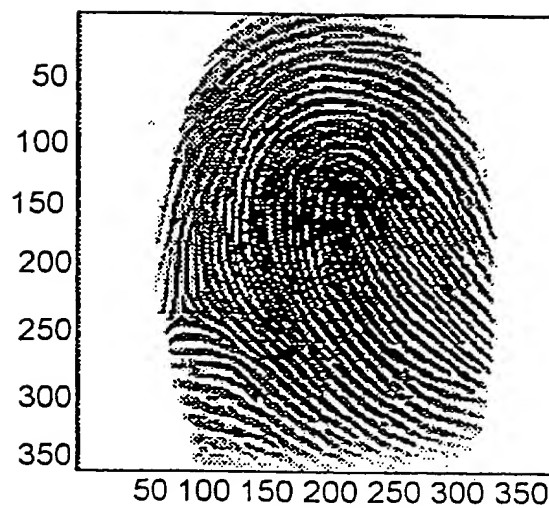


Fig. 48

THIS PAGE BLANK (USPTO)

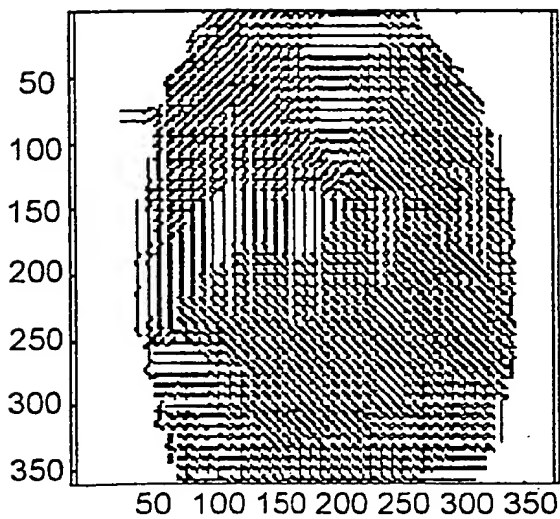


Fig. 49

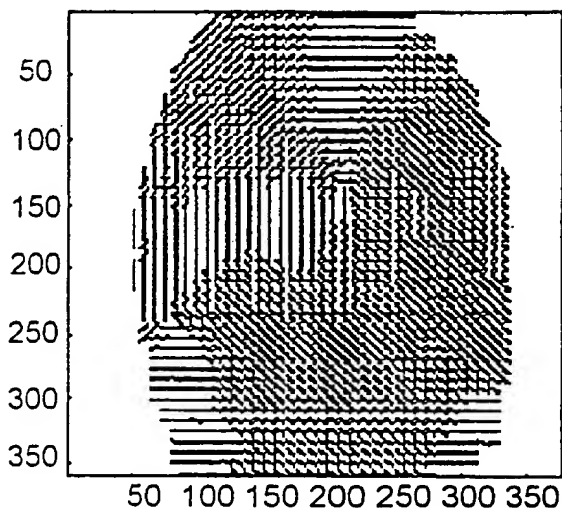


Fig. 50

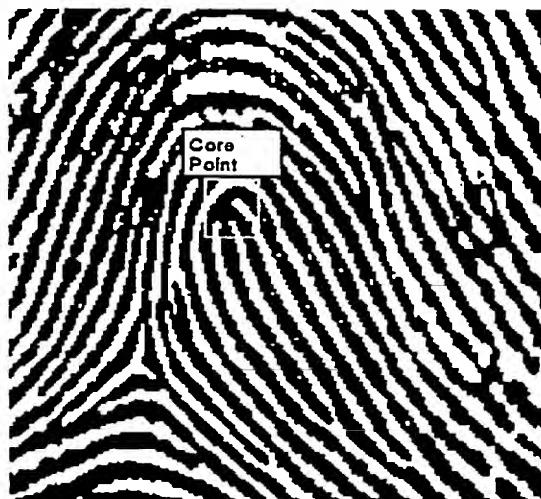


Fig. 51

THIS PAGE BLANK (USPTO)

20/27

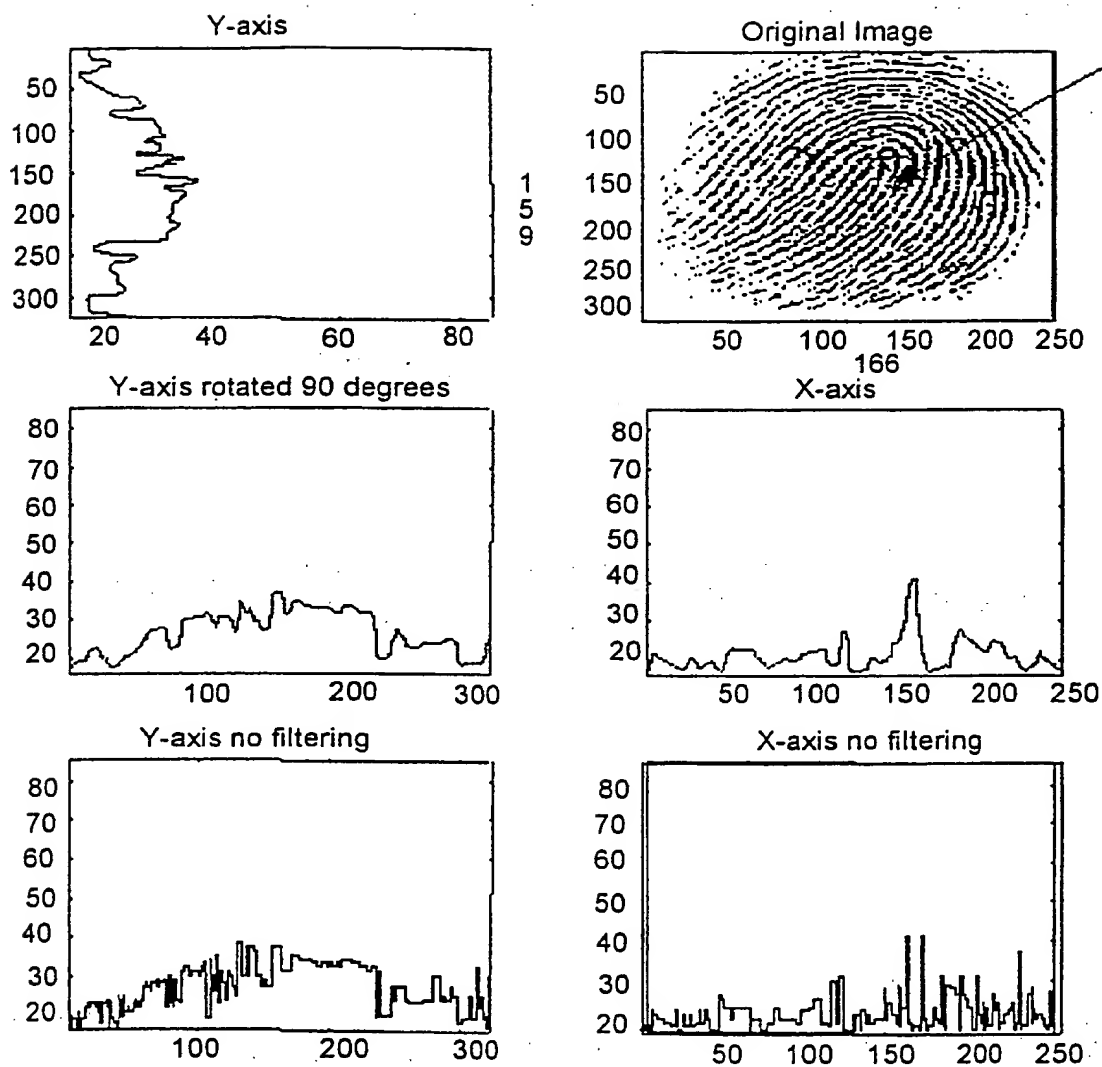


Fig. 52

THIS PAGE BLANK (USPTO)

21/27

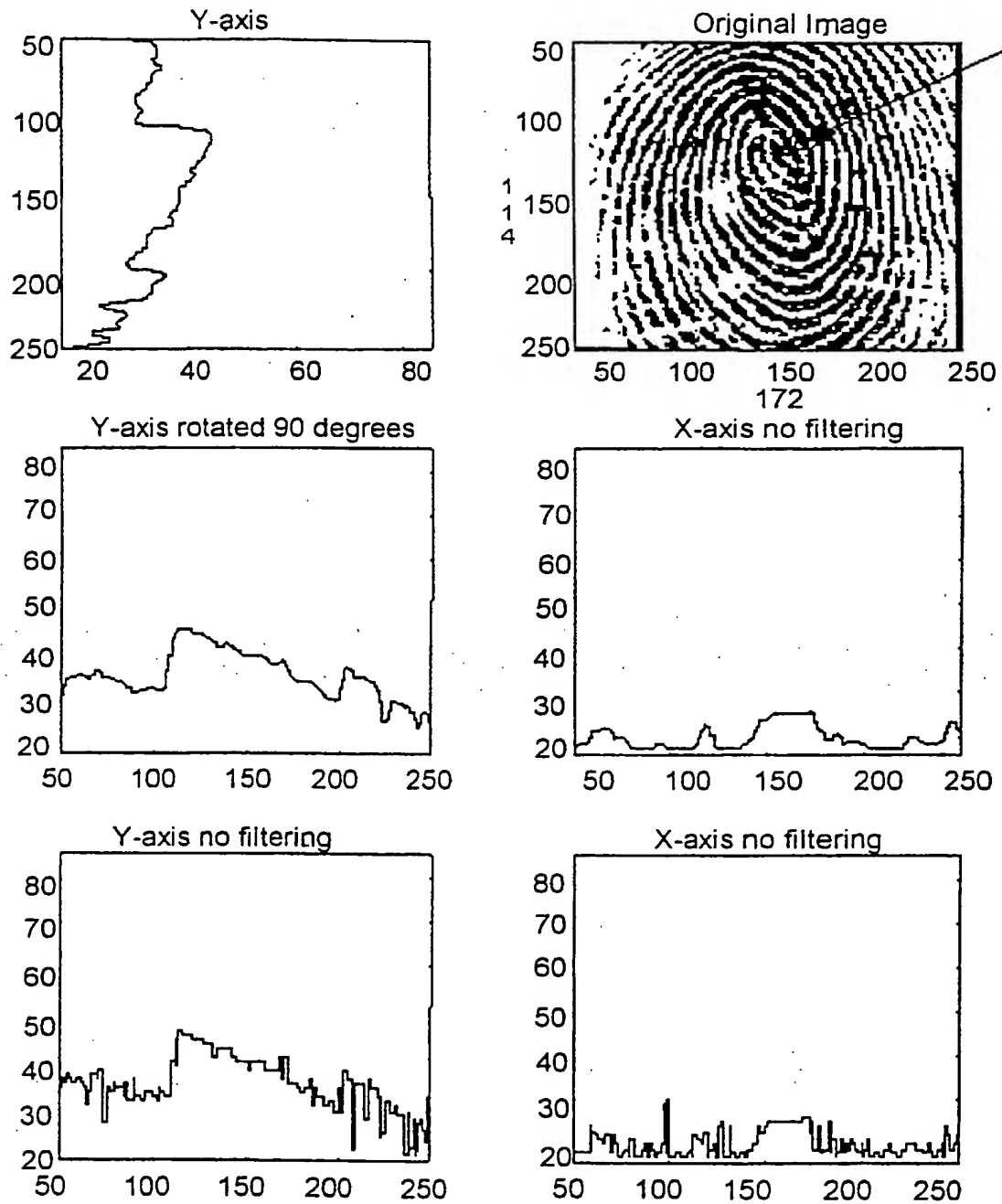


Fig. 53

THIS PAGE BLANK (USPTO)

22/27

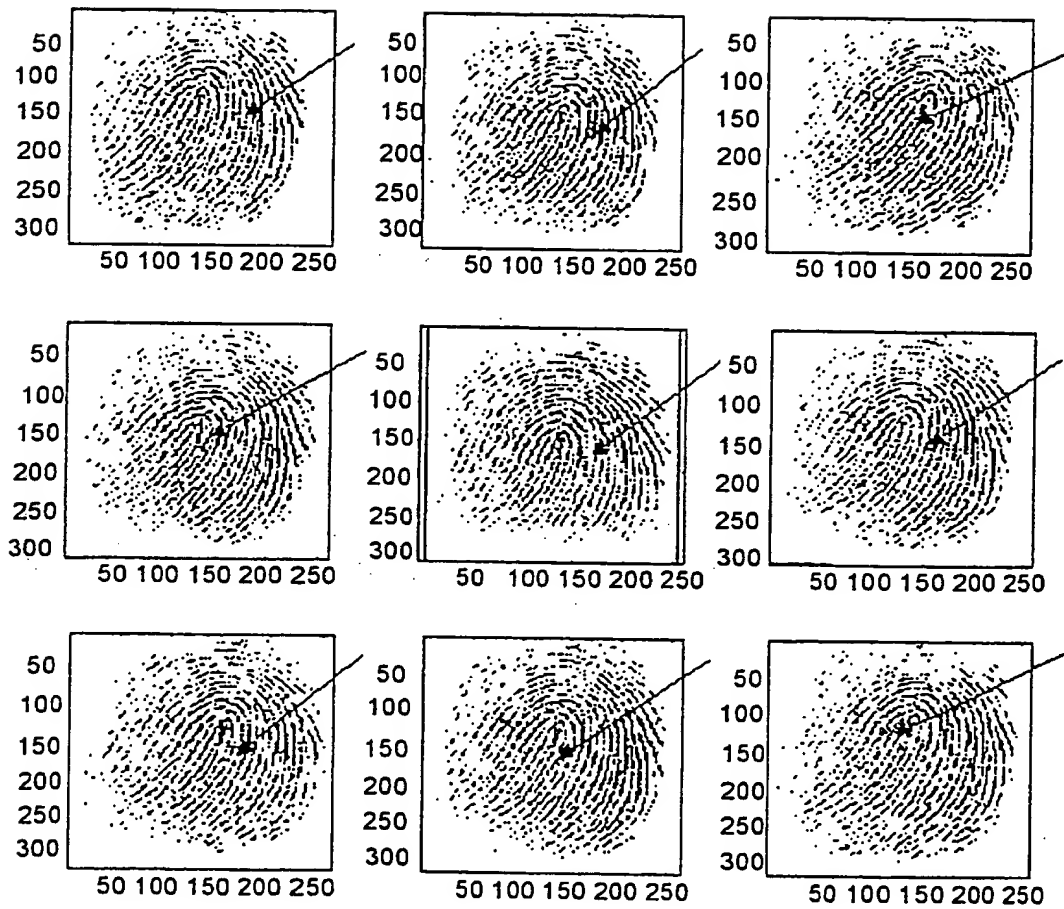


Fig. 54

THIS PAGE BLANK (USPTO)

23/27

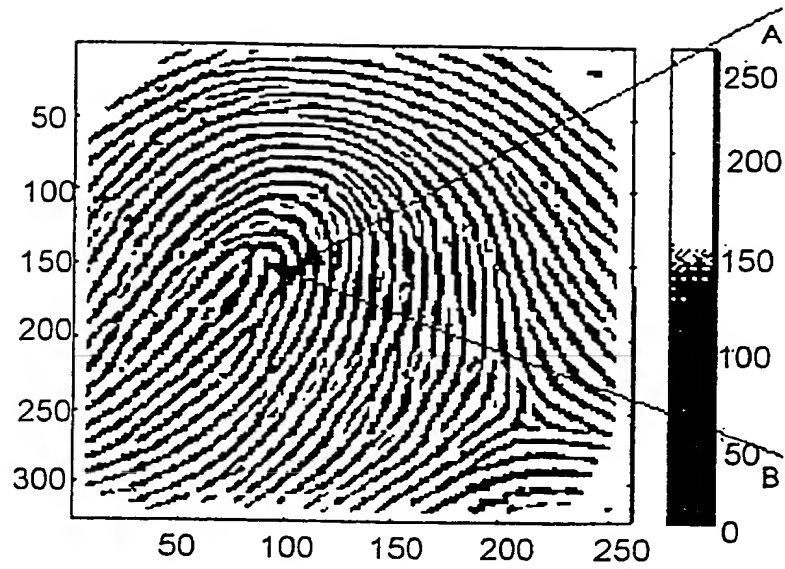


Fig. 55

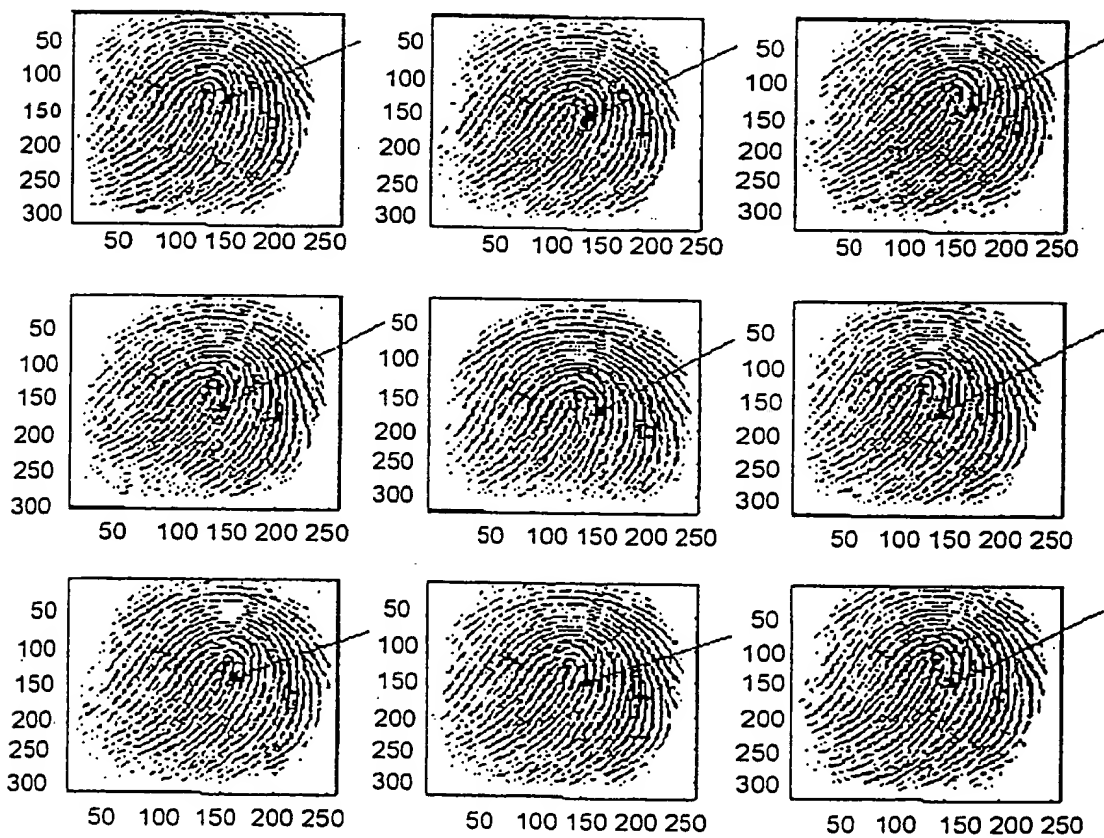


Fig. 56

THIS PAGE BLANK (USPTO)

24/27

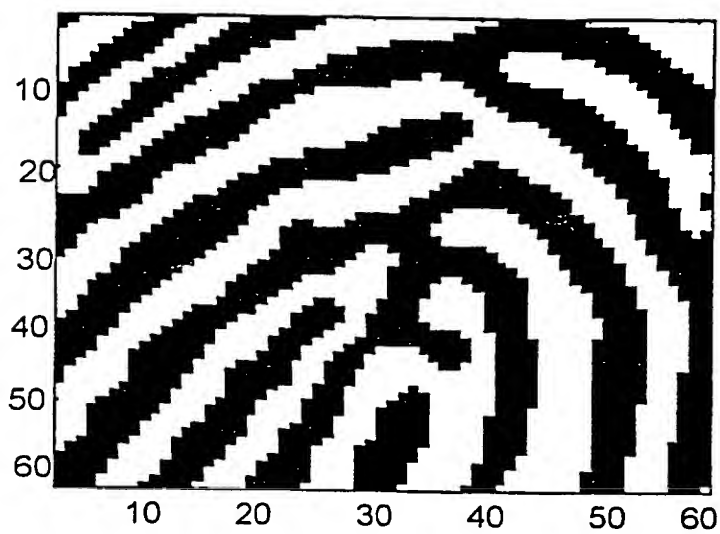


Fig. 57

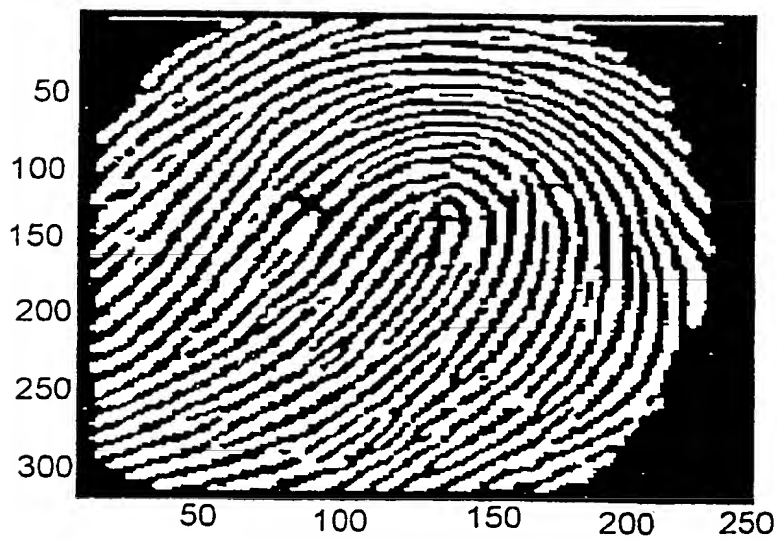


Fig. 58

THIS PAGE BLANK (USPTO)

25/27

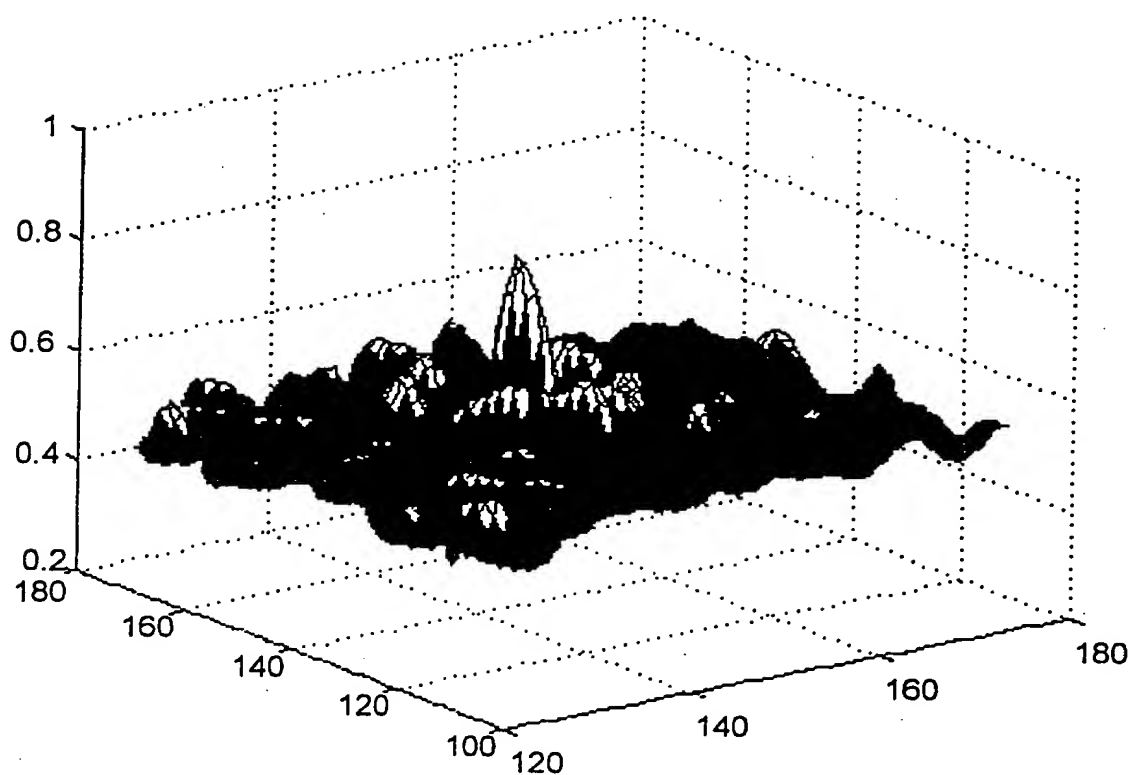


Fig. 59



Fig. 60

THIS PAGE BLANK (USPTO)

26/27

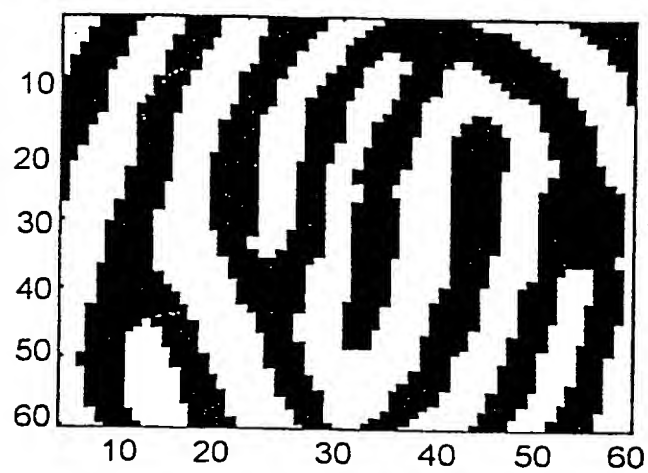


Fig. 61

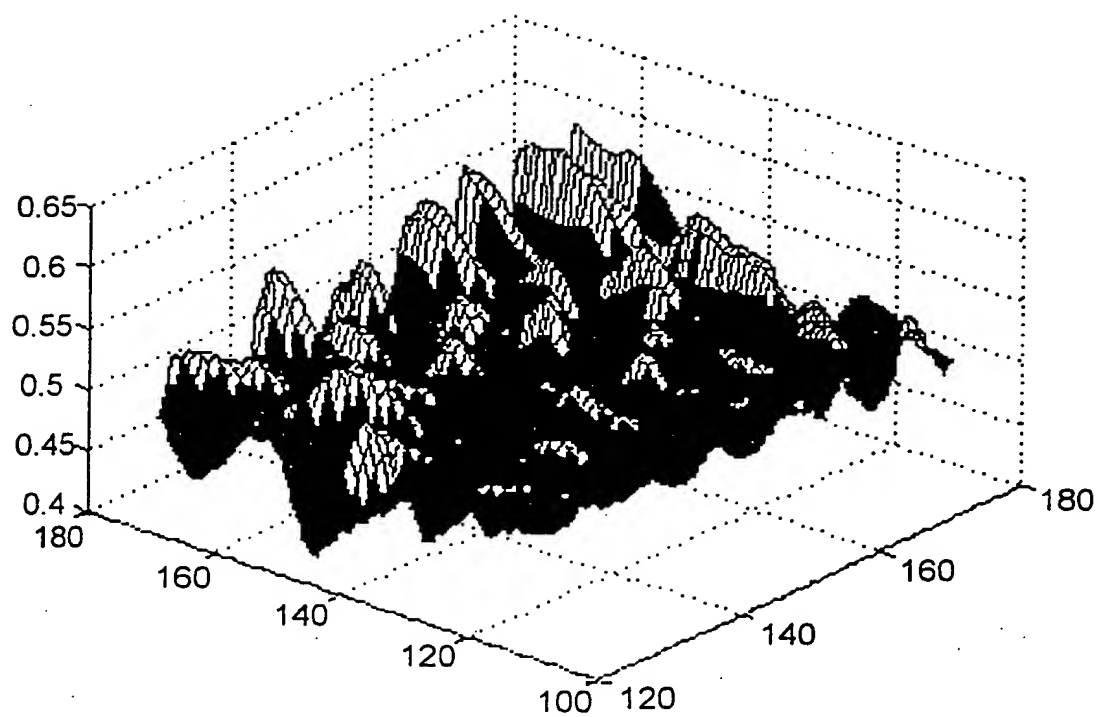


Fig. 62

THIS PAGE BLANK (USPTO)

27/27

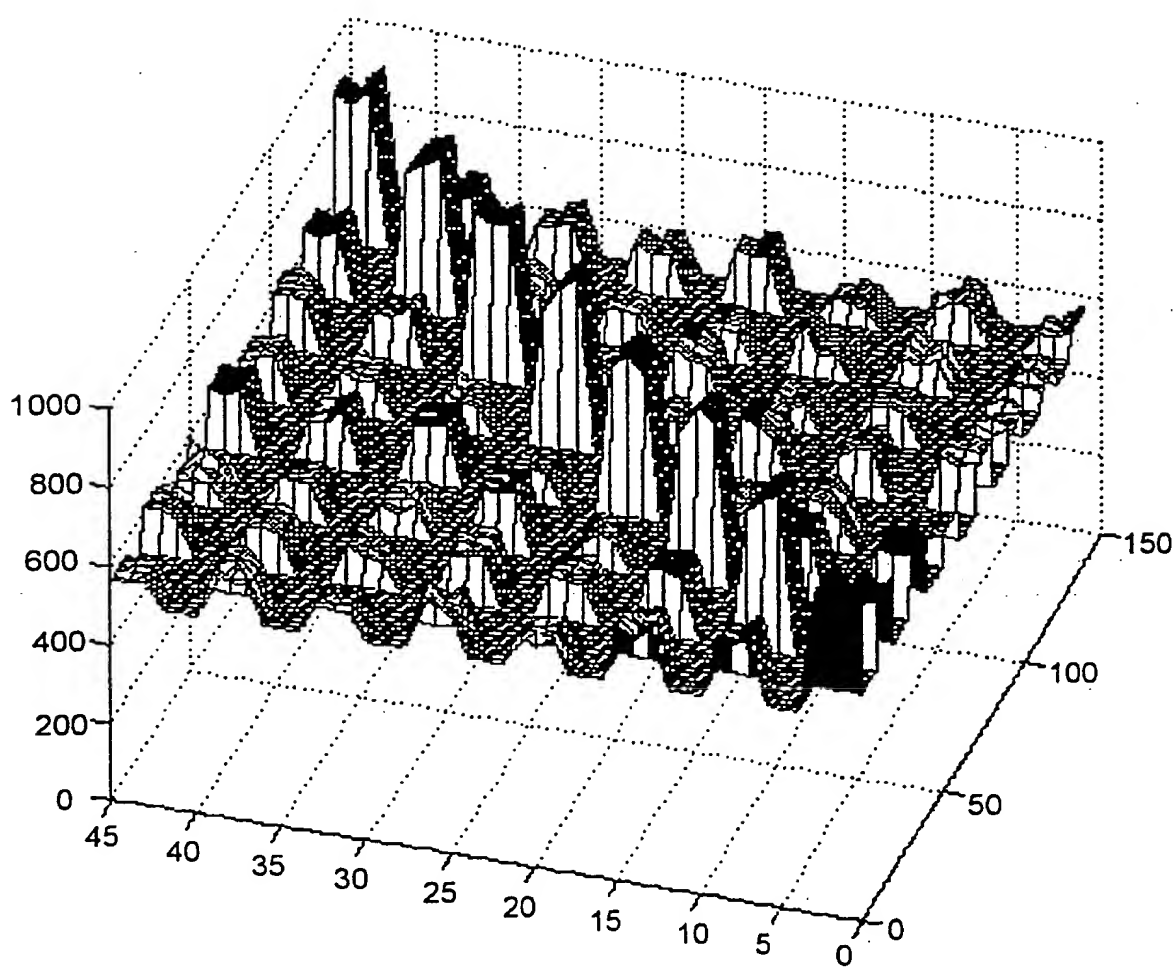


Fig. 63

THIS PAGE BLANK (USPTO)